No. 018

January - April 1969

COMPUTER-AIDED SHIP DESIGN
LECTURE NOTES

Horst Nowacki

THE DEPARTMENT OF NAVAL ARCHITECTURE AND MARINE ENGINEERING

THE UNIVERSITY OF MICHIGAN
COLLEGE OF ENGINEERING

COMPUTER-AIDED SHIP DESIGN

LECTURE NOTES

BY

HORST NOWACKI,
DR.-ING., ASSOCIATE PROFESSOR

JANUARY-APRIL 1969

DEPARTMENT OF NAVAL ARCHITECTURE AND MARINE ENGINEERING
COLLEGE OF ENGINEERING
THE UNIVERSITY OF MICHIGAN
ANN ARBOR, MICHIGAN

Motto: Design is an aim,
Computers are an aid.

PREFACE

This set of lecture notes was written for a new course in the curriculum of the Department of Naval Architecture and Marine Engineering: NA 574, "Computer-Aided Ship Design." This senior and graduate level course is to supplement existing courses in ship design and computer use. The new course therefore deliberately avoided any unnecessary duplication. The question it is intended to answer is rather: "How does one convert a ship design problem into one the computer can help to solve?"

It is also felt that in the professional environment of our industry there is a lot of competence in ship design and computer programming, but there still exists some sort of void between these two specialties. We want to educate engineers who have this dual competence, but also an understanding of the logical nature of design and the associated computer methodology.

The notes also attempt to give an overview of the current state of developments despite the realization that the available information is of necessity incomplete, and despite the threat of early obsolescence. But there seems to be some merit to taking a panoramic view, if only to realize how much alike problems and solutions are throughout the profession.

Examples and illustrations were kept scarce. But a set of lecture notes is only intended as the COMPILE-phase of a course program. The EXECUTE-operation is performed by the instructor and his class in the classroom and, hopefully, at home.

Ann Arbor, April 18, 1969 Horst Nowacki

Lecture Notes on Computer-Aided Ship Design.

Contents:

Page

1. Mathematical Representation of Hull Form

1.1 Historical introduction.

The idea to use simple geometric shapes such as circles, ellipses, parabolas, sine curves in ship design is probably an ancient one. Söding, Ref. 30, gives an account on numerous applications of such shape elements since the late middle ages, and he mentions that even in earliest known technical drawings of ships (Timbotta, 1445) one detects a sine shaped camber line construction.

The advantages of simple geometric shapes in all stages of ship production are obvious (surface development, fitting, erection). We must also keep in mind that the lines drawing was not invented before around 1700, and it was only then that the free design of empirical shapes and their accurate reproduction became feasible.

Aside from such direct benefits, it seems the use of mathematical ship lines was also motivated sometimes by a certain magic belief in the superior performance of hull forms containing particular geometric elements. It is hardly a coincidence, for example, that the age that had produced Gerstner's trochoidal wave theory (1802) also experimented with trochoidal ship lines, Ref. 32.

As far as we can tell from documented history, the first to apply mathematical shapes systematically in ship design was apparently the Swedish naval constructor Chapman who, in his book around 1760, Ref. 1, mentioned the use of a family of parabolas for waterlines and other curves on the hull surface.

But it was D. W. Taylor who, right after the turn of the century, began to use mathematical expressions to represent in a rational manner the empirical hull shapes of the present day, Ref. 2. He approximated the sectional area curve and the waterlines of the ship by fifth order polynomials whereas the sections were represented by fourth order polynomials or hyperbolas depending on fullness. For the sectional area curve, for example, he used the formulation

$$y(x) = a + bx + cx^2 + dx^3 + ex^4 + fx^5 \qquad (1)$$

and, in the nondimensional coordinates illustrated in Fig. 1, satisfied the conditions:

$$\left.\begin{array}{lll} \text{At} & x = 0 & y = 0 \\ & & y' = t, \text{ entrance slope, given} \\ & & \\ \text{At} & x = 1 & y = 1 \\ & & y' = 0 \\ & & y'' = \alpha, \text{ curvature at midship section, given} \\ & & \\ \text{Further:} & & \int_0^1 y(x)\,dx = p, \text{ forebody prismatic, given} \end{array}\right\} (2)$$

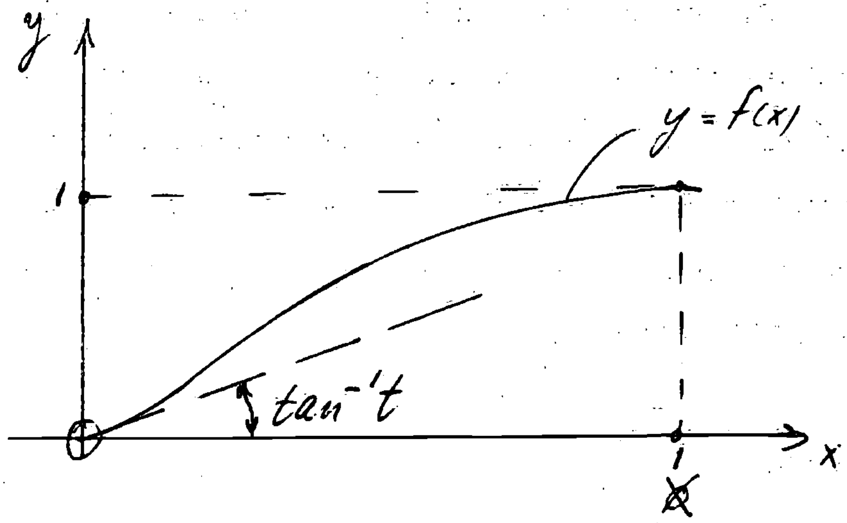


Fig. 1. Taylor's coordinates for sectional area curve and waterlines

This gave him six equations for the six unknown coefficients of Eq. 1.

Taylor applied this method to the British cruiser Leviathan, the parent of his systematic series, and then by systematic variation of the geometric constraints, Eq. 2, developed what became known as Taylor's Standard Series.

The realization that a systematic description and variation of empirical hull form is essential for a meaningful interpretation of the results of hydrodynamic research also motivated Weinblum in his work around 1930, Ref. 3. He developed a descriptive system for axisymmetric and two-dimensional shapes, and added much to the understanding of ship geometry and fairness.

For further details about the early history of mathematical ship lines up to about 1955, see Saunders, Ref. 32.

In the modern era, technological progress broadened the motivation for mathematical ship lines. The use of computers made it possible to extend mathematical hull representation to diverse practical applications in ship design and production such as lines fairing, and graphic display, lofting, and the control of automated machine tools. Lately, even the creation of lines from a few given design constraints has become an area of active experimentation.

1.2 Objectives of hull representation methods

The purpose of the mathematical representation of a hull may be:

*1. Given a graphical representation of the hull form (lines drawing) that is in agreement with the designer's intentions in every respect (fairness, accuracy, etc.). Find the closest possible mathematical approximation of this form. If in this process we use offsets lifted from the drawing we are led to a task known as interpolation.

*2. Given a table of offsets read from a small scale lines drawing that is not perfectly fair. Assume further that the table contains some bad points due to reading errors or mistranscription. Eliminate the bad points (smoothing), and obtain a hull representation that looks fair (see below) on the 1:1 scale (fairing). Nonetheless, retain the character of the original lines drawing as closely as possible.

*3. Given a very crude sketch of a lines plan, or only the parameters of the hull to be designed. Create a set of lines meeting the designer's intentions and refine it to the level where it is fair enough for every future purpose (Lines creation).

These tasks differ in the quality and scope of information of their inputs, but they all aim at an output that represents a fair ship. There are differences, though, as to the desired accuracy. High demands for production purposes, lofting, automatic tools. Simplified ship sufficient for many purposes in hydrodynamics.

The leeway for hull modifications increases from *1 to *3. In the case of *1 we often desire

"Practical identity"

of input and output, i.e., the computer is only permitted to negotiate those inaccuracies that are due the small scale of the drawing. "Do not change by more than the reading error"! (Drawing:0.005", full scale perhaps 0.25".)

Definitions of smoothness and fairness.

The terms "smoothness" and "fairness" are often used somewhat subjectively, and need to be defined in the present context to give them unique meaning.

Smoothness, for our purpose, must not be confused with the concepts of hydraulic smoothness (absence of small protuberances) or mathematical smoothness (continuous first derivative, no corners). We want to understand smoothness as the absence of local bumps. A "bump" consists of two closely spaced inflection points, Fig. 2. It may be due to inadvertent bad input points or poor fairing of the input lines. The bad data trouble ought to be removed by "smoothing" before fairing should start.

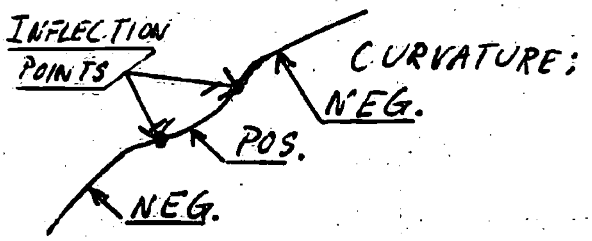


Fig. 2: Bump

The difference between intended inflection points such as in hollow waterlines and inadvertent ones (local bumps) can be signalled to the computer by providing enough input points and by the following agreement: A local bump is one that has two inflection points in the interval of, say, five data points. For more details, see Ref. 28.

Fairness is often used in the subjective sense of "pleasing to the eye." But, in addition, some necessary conditions can be formulated for the fairness of a curve (see for example Weinblum, Ref. 3):

*Continuous first derivative of the curve, no corners.

*Continuous second derivative (fairness to the second order).

Incidentally, in manual fairing by means of splines (battens) the two above fairness qualities are ensured automatically. Fig. 3 illustrates that the spline, considered as an elastic beam, has continuous bending moments within each interval and also where it is supported by the weights, and hence continuous second derivatives $(y''(x) \propto M(x))$. For more details, see section 1.3.

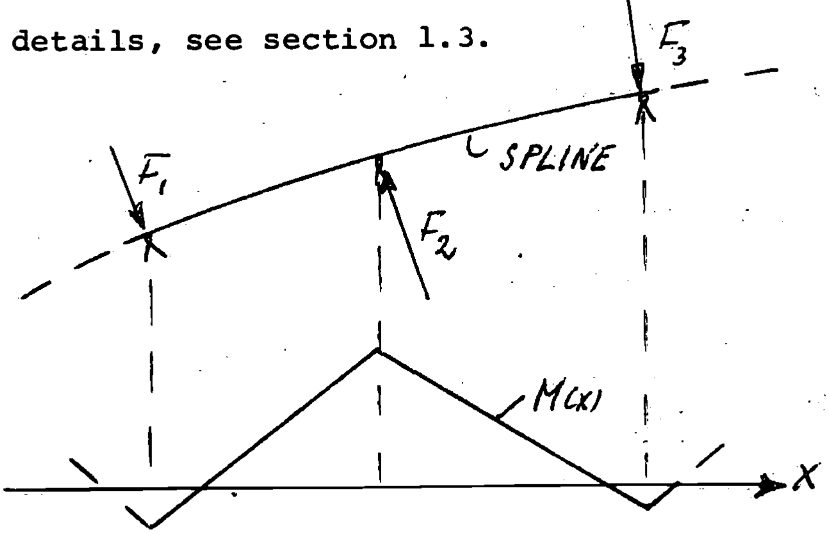


Fig. 3: Spline segment

*Continuous higher order derivatives ($y^{(k)}$, fairness to the k'th order). This type of fairness can hardly be fully realized, but may serve as a target.

*No flat parts amidst curved regions.

Other typical properties of ship lines, not necessarily deductible from the fairness quality, are:

*Few inflection points per quadrant.

*Usually monotonic character of curves. ?

*No negative offsets. [illegible]

1.3 Techniques of hull form representation.

In view of the many different purposes of hull representation, and the great variety of mathematical tools available, it is no wonder that a multitude of techniques has been developed, each advocated by its disciples for different, but by and large legitimate reasons.

Table I gives an overview of such techniques. Some of their main distinctive properties are listed in the table, namely:

*Interpolation vs. approximation methods: In an interpolation, the curves are fitted through fixed points, as apposed to an approximation by least squares or other criteria.

*Analytic representation of curve (surface) by single expression vs. same by several segments (patches).

*Draft function vs. sectional methods vs. direct surface fitting methods: In a draft function method, only waterlines or similar curves are given a direct mathematical representation, and points between such grid lines are obtained by draftwise interpolation. In the so-called sectional methods, all sections (waterlines, sections, buttocks, diagonals), or at least more than one set, are represented mathematically. Existing discrepancies are negotiated, usually by iteration, until a unique hull representation is obtained. Finally, instead of using such an iterative process applied to sets of curves, one may fit the whole surface simultaneously in a direct surface fit.

Table I: HULL REPRESENTATION METHODS.

No.	Author	Affiliation	Country	Date	Ref.	Purpose	Input	Procedure	Function	Section	Patches
1	D. W. Taylor	U.S. Navy	U.S.	1915	2	Creation and syst. variation	Hull param.	Draft function	Polyn.	1.32	2
2	Weinblum	Univ. of Berlin and Hamburg	Germany	1934,...	3,8	Syst. variation, classification	Hull param.	(Incompl.)	Polyn.	1.32	2
3	Benson		England	1940	4,22	Creation of lines	Hull param.	(Incompl.)	Polyn.	1.32	2
4	Lackenby	BSRA	England	1950	6	Syst. variation	Parent Hull	Affine distortion	—	—	No
5	Thieme	University of Hamburg	Germany	1952,56	7,9	Creation and classification	param.	(Incompl.)	Polyn.	1.32	2
6	Taggart	Reed Research	U.S.	1955	10	Creation	Hull param.	(Incompl.)	Polyn.	(1.32)	2
7	Theilheimer & Starkwheather	U.S. Navy	U.S.	1957,61	11,17	Interpolation and/or fairing	Offsets	Draft function	Discont. cubics	1.31	2
8	Rösingh & Berghuis	Wilton-Fijenoord, Bull Ned.	Holland	1959	12	Fairing	Offsets	Draft function	Arbitrary	1.34	3
9	Pien	U.S. Navy	U.S.	1960	13	Approxim.	Offsets	sectional method	Polyn.	1.34	2

Table I, cont: HULL REPRESENTATION METHODS.

No.	Author	Affiliation	Country	Date	Ref.	Purpose	Input	Procedure	Function	Section	Patches
10	Martin	NPL	England	1961	14	Rough approxim.	Offsets of S.A. curve	(Incompl.)	Cheb. polyn.	(1.39)	2
11	Kerwin	MIT	U.S.	1960	15	Rough approxim.	Offsets	Sectional method	Legendre polyn.	1.33	No
12	Lidbro	(Linköping)	Sweden	1961	16	Interpolation	Offsets	surface fitting	Polyn. & some others	1.35	Many
13	F. Taylor	King's College, Durham	England	1962	19	Interpolation	Waterline Offsets	(Incompl.)	Cheb. polyn.	(1.33)	2, or 3
14	Miller and Kuo	Univ. of Glasgow	Scotland	1963	20	Interpolation	Offsets	Draft function	Polyn.	1.32	Several
15	Berger, Webster...	Todd Shipyards	U.S.	1963,66	21,28	Fairing	Offsets	Surface fitting	Discont. cubics	1.31	Yes
16	Williams	SSET, Gothenburg	Sweden	1964	23	Creation of lines	Hull parameters	Draft function	Polyn.	1.32	Yes
17	Hamilton and Weiss	MIT	U.S.	1964	24,40	Creation of lines	Hull parameters	Surface fitting	Surface cubics	1.35	Many
18	Bakker	NSMB	Holland	1965	25	Fairing	Offsets	Sectional method	(Not needed)	1.31	Yes

Table I, cont: HULL REPRESENTATION METHODS.

No.	Author	Affiliation	Country	Date	Ref.	Purpose	Input	Procedure	Function	Section	Patches
19	Gospodnetić	NRC, Ottawa	Canada	1965	26	Interpolation	Offsets	Sectional method	Elliptic integrals	1.31	Yes
20	Corin	U.S. Navy	U.S.	1966	27	Fairing	Offsets	Sectional method	Discont. cubics	1.31	Yes
21	Tuck, and V. Kerczek	U.S. Navy	U.S.	1968	29	Fairing	Offsets	Sectional method	Conformal mapping	1.34	No
22	Söding	T H Hanover	Germany	1966	30	Creation and interpolation	Offsets	Sectional method	Discont. polyn.	1.31	Yes
23	Kantorowitz	DSRI, Lyngby	Denmark	1967	31	Interpolation	Offsets	Surface fitting	Orthogonal polyn.	1.35	Several
24	Kaiser, et al.	Dornier, Friedrichshafen	Germany	1968	41	Interpolation	Offsets	Surface fitting	Surface polyn.	1.35	Many
25		Bergens Mek. Verkst.	Norway	~1961	—	Fairing	Offsets	Draft function	Polyn.	—	5
26	(AUTOKON)	CIIR, Oslo, Aker Group	Norway		—	Fairing	Offsets	Sectional method	Spline polyn.	—	Several

*Simple polynomials (cubics) vs. orthogonal polynomials vs. other functions.

*According to the mathematical tools involved:

Spline methods

Parametric methods

Direct polynomial methods

Transformation methods

Surface fitting methods

Illustrations for each of these categories are given in the following.

1.31 <u>Spline methods</u>.

1.311 <u>Spline interpolation</u>.

Let us consider the problem of finding a curve through a given set of points in a manner analogous to manual spline fairing. In Fig. 4, we show the i'th segment of the curve we intend to find, the segment ends being defined by the location of the spline weights (ducks) we have arranged at the given offset points to ensure an exact interpolation through these points. Let there be N such segments, or N+1 offsets. The line may represent a waterline or any other curve that can be represented in an x-y space. The ducks are assumed to give ideal, frictionless support so that the spline is simply supported.

We want to determine mathematical expressions for the spline curve. This problem has been attacked by Theilheimer and Starkweather, Refs. 11 and 17, and has later on been taken up by many others, Refs. 21, 26, 27, 28, 30, 35. The following

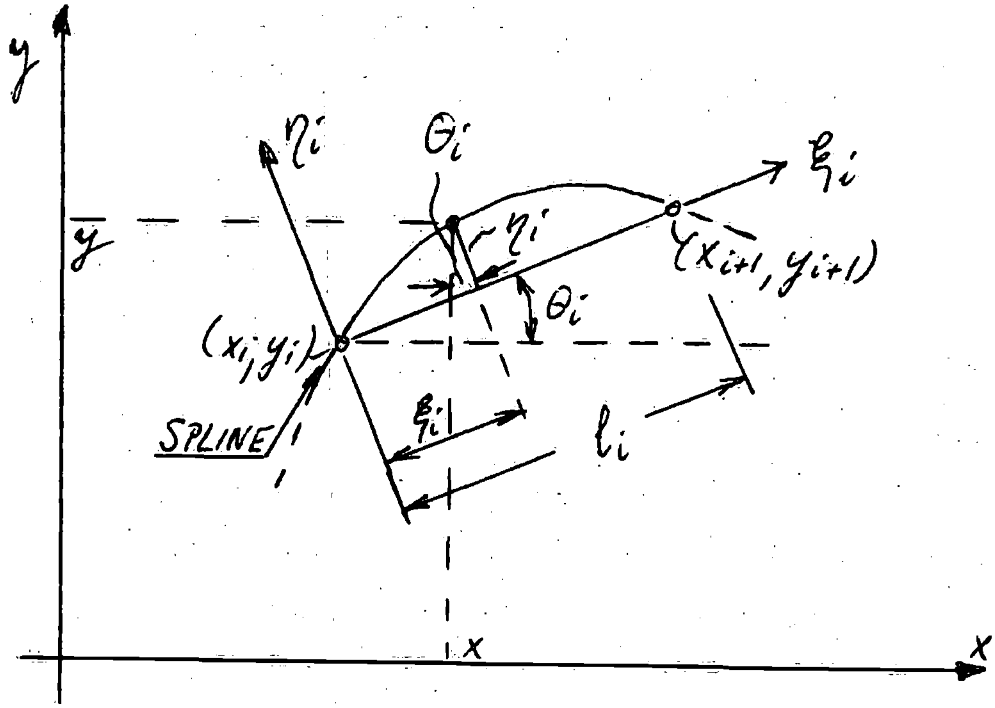


Fig. 4: Coordinate systems for spline interpolation

adopts much of Johnson's approach, Ref. 36.

Let us, first of all, remember that the simple supports can only transmit shear forces. The bending moments are therefore continuous at the supports, and, due to the absence of external loads, vary linearly between supports.

In the $\xi_i - \eta_i -$ coordinate system of the i'th element:

$$M(\xi_i) = M(0) + B_i \xi_i \qquad (3)$$

B_i, a constant.

The supporting points can generally be arranged so that the maximum deflection in the field, $\eta_{i_{MAX}}$, is small relative to the length of the field, l_i :

$$\eta_{i_{MAX}} \ll l_i$$

This allows us to use elementary (linearized, Euler-Bernoulli) beam theory to formulate the spline deflection problem, neglecting $\eta_i'^{2}(\xi) \ll 1.$ Let us further assume for the time being that we use a constant cross section spline. Then we may write:

$$E\,I \cdot \eta''(\xi_i) = M(\xi_i) \tag{4}$$

$E =$ Hooke's modulus of elasticity

$I =$ spline area moment of inertia

Substituting (3) into (4) and integrating twice, we obtain (Ref. 36):

$$EI\,\eta(\bar{\xi}) = \int_0^{\bar{\xi}}\int_0^{\bar{\xi}} (M(0) + B_i \cdot \xi)\, d\xi =$$

$$= EI\Big(\eta(0) + \eta'(0)\,\bar{\xi} + \tag{5}$$

$$+ \frac{M(0)}{2EI}\,\bar{\xi}^{2} + \frac{B_i}{6EI}\,\bar{\xi}^{3}\Big)$$

where $\bar{\xi}$ = internal point of interval.

The essential conclusion is that the spline curve assumes the form of a cubic polynomial. Cubics are thus the simplest functions suitable for spline interpolation.

We shall discuss later how to determine the unknown constants in (5). It is clear from the foregoing that every interval has its own cubic, represented in its local coordinate system. Will it be possible to represent each cubic as a polynomial in the x-y-system, $y = f(x)$, and to combine all of these polynomials into a single analytic expression?

For this purpose we have to transform the cubics from the $\xi_i - \eta_i$, to the $x-y$ coordinate system (or any other unique coordinate system, for that matter). The transformation will generally consist of a translation and rotation. The shape of the curve will not be affected since it is invariant to such a transformation. Let the curve in the i'th interval be given by the third order polynomial:

$$\eta_i = P_i(\xi_i) \qquad (6)$$

According to Fig. 4, ξ_i and η_i transform into

$$\left.\begin{aligned} x &= x_i + \xi_i \cos\theta_i - \eta_i \sin\theta_i \\ y &= y_i + \xi_i \sin\theta_i + \eta_i \cos\theta_i \end{aligned}\right\} \qquad (7)$$

Substituting (6), this will be of the form

$$x = P_{1i}(\xi_i, \theta_i) \qquad (8a)$$

$$y = P_{2i}(\xi_i, \theta_i) \qquad (8b)$$

where P_{1i} and P_{2i} are two <u>different</u> third order polynomials in ξ_i, depending on the parameter θ_i.

It will be impossible to represent the spline curve in the form

$$y = P_3(x) \qquad (9)$$

where P_3 a cubic, because when substituting (8a) and (8b) into (9) the powers of ξ_i on the right-hand side in P_3 will exceed those for y on the left, which leads into contradictory conditions for the coefficients of $P_3(x)$. This means the spline curve will have to assume a parameter form like (8a), (8b) in the $x-y$ system. Incidentally, the same reasoning is valid whatever the order of the original polynomial $P_i(\xi_i)$ because the polynomials P_{1i} and P_{2i} are always of the same order.

It might still be desirable to combine the parametric expressions for x, and y, respectively, so as to obtain two polynomial expressions valid throughout the curve. But this is ruled out in general because the third derivatives are discontinuous for spline curves at the supports due to the duck reactions whereas polynomials are continuous in all derivatives.

In summary, neither can spline curves be transformed into polynomials of the form $y=f(x)$ for the whole curve, nor can such polynomials be used to represent exactly ship lines generated by splines.

In many variants of the spline method, particularly in Refs. 17, 21, 26, 27, 28, the curve is approximated by a cubic in the $x-y$ coordinate system rather than in the $\xi_i - \eta_i$ systems. This avoids the parameter representation, but reduces the applicability of elementary beam theory because it is more difficult to satisfy the slope requirement $y'^2 \ll 1$ than that of $\eta_i'^2 \ll 1$. Nonetheless practically useful results have been obtained. In addition, these methods have succeeded in obtaining a closed form analytic expression for the whole curve despite the discontinuities of the third derivatives by introducing a special notation for discontinuous polynomials. The spline curve through the points $x_0, \dots, x_N$ is written as

$$y = f(x) = a + bx + cx^2 + A_0 x^3 + A_1 (x - x_1)^3_{\#} + \dots + A_{N-1} (x - x_{N-1})^3_{\#} \qquad (10)$$

y = Theilheimer polynomial

where the terms

$$\left.\begin{aligned} (x - x_n)^3_{\#} &= 0 \quad \text{for } x \le x_n \\ &= (x - x_n)^3 \quad \text{for } x \ge x_n \end{aligned}\right\} (10a)$$

provide the discontinuities.

In the above case of Eq. 10 there are N + 1 given offsets, and N + 3 unknown coefficients. This corresponds to the fact that we have not yet decided how we want to support the spline outside the interpolation range. In the practical determination of the coefficients in (10), we therefore get N + 1 equations from the given offsets which are of the type

$$y_n = f(x_n) \quad , \quad n = 1, \dots, N-1 \qquad (11)$$

and we must make two further assumptions, for example so as to minimize the sum of the squares of the discontinuities in the third derivatives, see Ref. 17. This results in a system of N + 3 normal equations for the coefficients. For more details about the form of this system and for an efficient solution technique, see Johnson, Ref. 36.

The "Exact" Spline.

D. Gospodnetić, Ref. 26, has developed a spline interpolation method based on the exact expression for the curvature (for a beam of finite deflection) rather than the linearizing assumption $\eta'^2 \ll 1$. In the coordinate system $\xi - \eta$ of Fig. 5 we obtain for the curvature of the spline segment shown:

$$\frac{1}{\rho} = \frac{\eta''}{\sqrt{1+\eta'^2}^{\,3}} = \frac{M}{EI} \qquad (12)$$

where

ρ = radius of curvature = $d\zeta/ds$

ξ, η = abscissa and ordinate in local coordinate system, with ξ in the direction of the bearing force P_0 ;

$\zeta = \eta' = \frac{d\eta}{d\xi}$ = slope of curve, see Fig. 5;

s = coordinate measured along arc from ξ-η origin;

I = spline moment of inertia, assumed constant;

$M = M_L + P(\eta_L - \eta)$ = bending moment;

M_L = end moment at far end;

$P = P_0 = P_L$ = magnitude of bearing forces;

η_L = ordinate of far end.

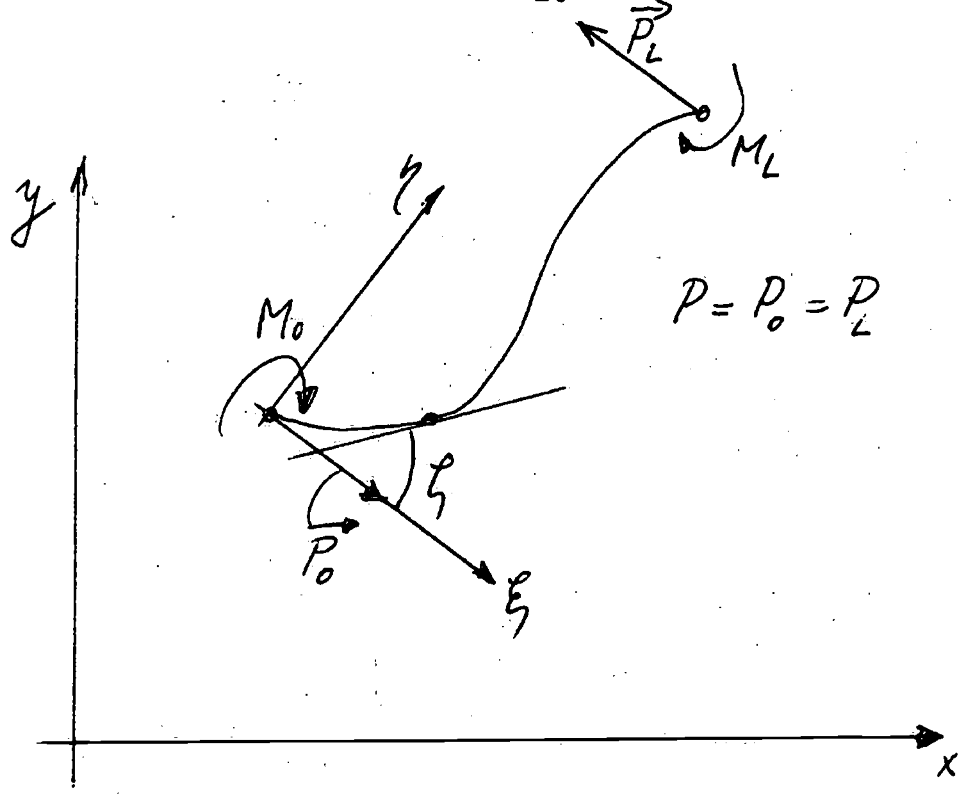


Fig. 5: Coordinate systems for exact spline, from Ref. 26.

Differentiating Eq. 12, and nondimensionalizing:

$$L^2 \frac{d^2\varphi}{ds^2} = -\beta^2 \frac{d\eta}{ds} = -\beta^2 \cdot \sin\varphi \qquad (13)$$

where L = arc length between supports

$$\beta = L\sqrt{\frac{P}{EI}}$$, a parameter

Multiplying by $d\varphi/ds$, and integrating we get

$$L^2 \int \frac{d\varphi}{ds}\, d\left(\frac{d\varphi}{ds}\right) = -\beta^2 \int \sin\varphi\, d\varphi + C \qquad (14)$$

$$L^2 \left(\frac{d\varphi}{ds}\right)^2 = 4\beta^2\left(C_1 - \sin^2(\varphi/2)\right) \qquad (15)$$

It can be shown, Ref. 26, that

$$\left(\sin^2(\varphi/2)\right)_{Max} \leq C_1 < 1 \qquad (16)$$

The following transformation will lead to elliptic integral expressions for the beam deflection. Let

$$\left.\begin{array}{l} C_1 = k^2, \qquad 0 \leq k < 1 \\ \sin(\varphi/2) = k \sin\psi, \quad \psi = \psi(s), \quad |\varphi| < \pi \end{array}\right\} \quad (17)$$

or by differentiation

$$\frac{1}{2} \cos\varphi \, d\varphi = k \cos\psi \, d\psi \qquad (18)$$

With proper sign conventions for curvature and slope, Ref. 26, one derives from (15) and (17)

$$\frac{d\varphi}{ds} = 2 \frac{\beta}{L} \sqrt{C_1 - k^2 \sin^2\psi} = 2 \frac{\beta}{L} \cos\psi \qquad (19)$$

and from (15), (17) and (18)

$$\left.\begin{array}{l} \dfrac{d\psi}{ds} = \dfrac{d\varphi}{ds} \dfrac{d\psi}{d\varphi} = \dfrac{d\varphi}{ds} \cdot \dfrac{\cos(\varphi/2)}{2k\cos\psi} = \dfrac{\beta}{L} \cos(\varphi/2) \\ = \dfrac{\beta}{L} \sqrt{1 - k^2 \sin^2\psi} \end{array}\right\} \quad (20)$$

The coordinates of the curve are defined by the elements

$$\frac{\beta}{L} d\xi = \frac{\beta}{L} \cos\varphi \, ds = \frac{\beta}{L}\left(2\cos^2(\varphi/2) - 1\right) ds \qquad (21)$$

$$\frac{\beta}{L} d\eta = \frac{\beta}{L} \sin\varphi \, ds = \frac{\beta}{L}\left(2\cos(\varphi/2) \cdot \sin(\varphi/2)\right) ds \qquad (22)$$

Substituting from (18)

$$\frac{\beta}{L} d\xi = \frac{1}{\sqrt{1 - k^2\sin^2\psi}} \left(2(1 - k^2\sin^2\psi) - 1\right) d\psi \qquad (23)$$

$$\frac{\beta}{L} d\eta = 2k \sin\psi \, d\psi \qquad (24)$$

Finally, by integration:

$$\left.\begin{aligned}\frac{\beta}{L}\xi &= \int_{\psi_0}^{\psi}(\ldots)d\psi = \int_{0}^{\psi}(\ldots)d\psi - \int_{0}^{\psi_0}(\ldots)d\psi = \\ &= 2E(k,\psi) - 2E(k,\psi_0) - F(k,\psi) + F(k,\psi_0)\end{aligned}\right\} \quad (25)$$

$$\frac{\beta}{L}\eta = 2k(\cos\psi_0 - \cos\psi) \qquad (26)$$

where

ψ_0 = value of ψ for $s=0$, see Ref. 26

ψ = variable defined in Eq. 17

$$F(k,\psi) = \int_0^{\psi}\frac{d\varphi}{\sqrt{1-k^2\sin^2\varphi}}$$ = incomplete elliptic integral of the first kind (27)

$$E(k,\psi) = \int_0^{\psi}\sqrt{1-k^2\sin^2\varphi}\,d\varphi$$ = incomplete elliptic integral of the second kind (28)

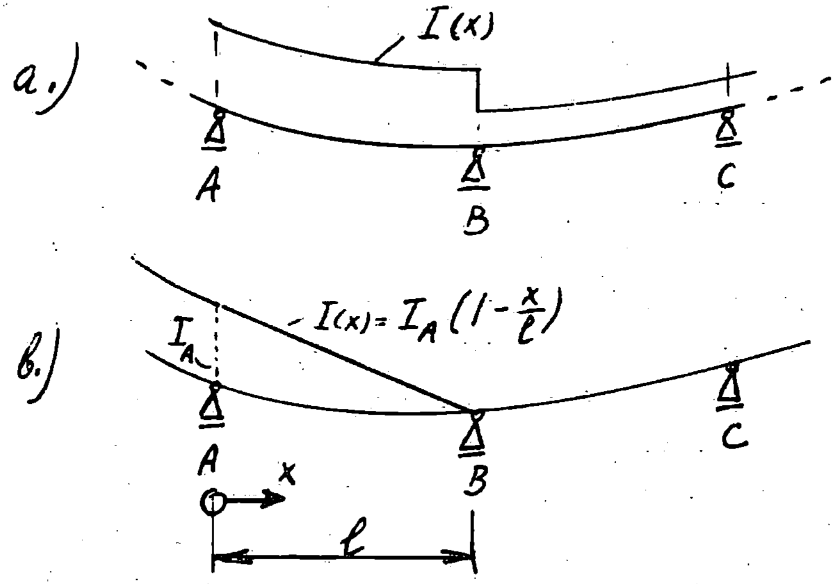
The conclusion is that the spline curve for arbitrary slopes and deflections may be expressed in terms of elliptic integrals. These integrals are easily computable by series expansion, see Refs. 26 and 38. The "elliptic parameters" of the spline curve k, ψ_0, and ψ, have to be determined from the boundary conditions of the segments analogous to the procedures in the polynomial spline methods. For details see Ref. 26. Once these parameters are known the spline interpolation by means of Eqs. 25, 26 is a simple task.

The Variable Cross Section Spline.

In manual fairing, one often encounters the problem that the spline cannot follow certain intended abrupt changes of curvature, for example at the transition into the parallel middlebody, or from flat bottom and sides into the bilge. One then reverts to piecewise fairing with French curves, or sometimes to variable cross section splines.

The mathematical splines of constant cross section, discussed up to this point, are equally unsuitable to produce sharp or even discontinuous changes of curvature. This is why many fairing techniques take exception of such curvature transitions, and subdivide the ship into patches so that the faired ship line does not have to cross a line of discontinuous or close to discontinuous curvature.

But there is a more elegant solution to this problem, the variable cross section spline technique pursued by Söding in Ref. 30. It is known that beams whose moments of inertia vary, for example as shown in Fig. 6, have discontinuous curvature at support B.



$$\eta'' = \frac{M}{EI} = \frac{M(1-\frac{x}{l})}{EI_A(1-\frac{x}{l})} =$$

= const.
in interval A-B

Fig. 6: Variable cross section splines

Söding finds it advantageous to use the spline of linearly varying moment of inertia, case b, wherever needed. The analysis of the spline deflection curve does not seem easy for this curve, but it turns out that one can use the well-developed solution techniques of structural analysis for continuous, multiply-supported beams. The analogy is apparent when we write:

<u>For the spline:</u>

$$\left.\begin{aligned}\frac{1}{l_{i-1}}\alpha_{i-1}+2\left(\frac{1}{l_{i-1}}+\frac{1}{l_i}\right)\alpha_i+\frac{1}{l_i}\alpha_{i+1} &= \\ =\frac{2(\gamma_{i-1}-\gamma_i)}{l_{i-1}}+\frac{\gamma_i-\gamma_{i+1}}{l_i}&,\ i=1,\ldots,N-1\end{aligned}\right\}(29)$$

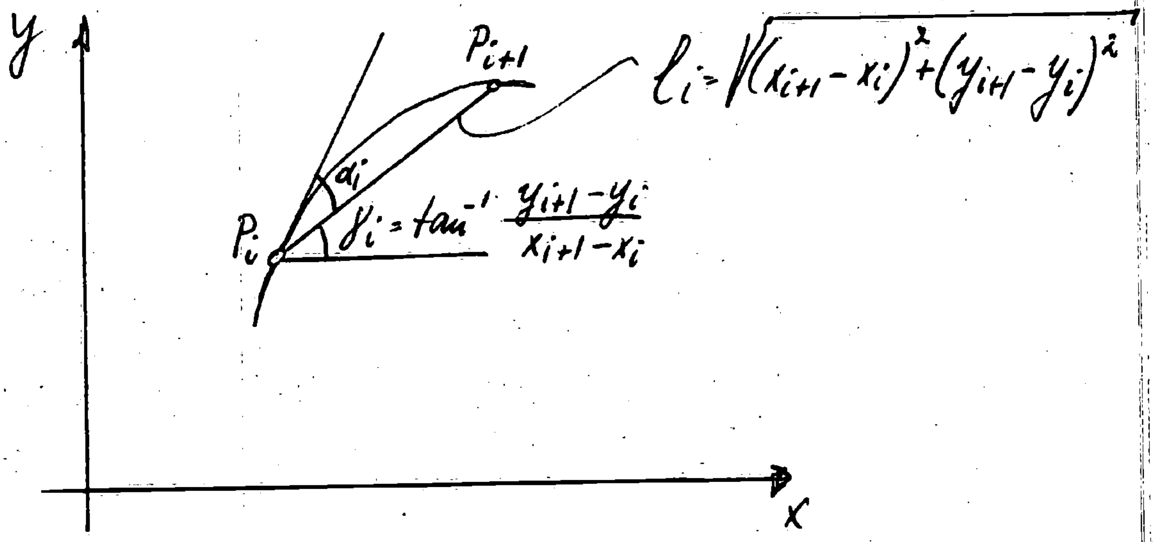
where

i = segment subscript

γ_i = slope of chord line, Fig. 7

l_i = length of chord, Fig. 7

α_i = angle between spline tangent at P_i, and chord line, Fig. 7



y
P_{i+1}
$l_i = \sqrt{(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2}$
α_i
$\gamma_i = \tan^{-1}\frac{y_{i+1}-y_i}{x_{i+1}-x_i}$
P_i
x

Fig. 7: Definition of angles and length

For three consecutive spans of a continuous beam (Three Moment Equation, Ref. 39):

$$\left.\begin{aligned} &l_{i-1}\cdot M_{i-1}+2(l_{i-1}+l_i)M_i+l_i M_{i+1}= \\ &=-l_{i-1}\cdot K_{i-1,r}-l_i\cdot K_{i+1,l}\,,\quad i=1,\dots,N-1 \end{aligned}\right\}(30)$$

where

M_{i-1}, M_i, M_{i+1} = bending moments

l_{i-1}, l_i = span lengths

$K_{i-1,r}, K_{i+1,l}$ = coefficients, depending on loads

In the analogy, the unknowns α_i and M_i are corresponding quantities, and the two equations are of exactly analogous form so that the systems of equations one can derive have matrices of identical form. Söding exploited this property in adapting a very efficient technique of structural analysis (a variant of the fixed-point method) to the fairing problem. His results are indeed convincing. For details see Ref. 30.

1.312 Spline Fairing.

Whenever the input data to a ship line interpolation contain "bad" data points, either due to actual mistakes or to reading inaccuracies, it is inevitable that some unwanted oscillations occur in the interpolated curve. To meet such a failure two things can be done in principle:

*The bad points may be eliminated or corrected, and another interpolation attempted. While this is good practice in filtering out those initial offsets that are obviously in error (smoothing, see below), it can hardly be done with success where the small, randomly distributed inaccuracies in lifting offsets from a drawing are concerned.

*The requirement for the curve to pass through all given points exactly may be dropped, and an approximate fit instead of an interpolation may be attempted. This leads to the fairing problem we want to discuss.

Least squares fairing.

The task of fitting an approximate curve through a given set of points is by no means unique, the result depending on the approximation criteria we select. The subject can probably best be introduced by studying the least squares fairing technique employed by Theilheimer and others in Refs. 17, 27.

Generally speaking there are two objectives in fairing. We want to ensure small deviations between the approximate curve, f(x), and the given offset points, y_n, hence

$$|f(x_n) - y_n| = \text{Min.}, \text{ for } n = 0, \ldots, N \qquad (31)$$

We also want to obtain a curve that shows only the intended inflection points, and no unwanted ones. If input points are given in sufficient density, the desired inflection points may be deduced from the second divided differences, defined for the n'th station as

$$r_n = \frac{\dfrac{y_{n+1} - y_n}{x_{n+1} - x_n} - \dfrac{y_n - y_{n-1}}{x_n - x_{n-1}}}{(x_{n+1} - x_{n-1})/2} \qquad (32)$$

or, for constant spacing, $h = x_{n+1} - x_n = x_n - x_{n-1}$,

$$r_n = \frac{y_{n+1} - 2y_n + y_{n-1}}{h^2} \qquad (33)$$

Wherever two neighboring second differences have the same sign, we conclude no inflection point is intended, and vice versa.

To obtain good agreement between curve, f(x), and given data in this respect, we want to minimize

$$\left| f''(x_n) - r_n \right| = \text{Min.}, \text{ for } n = 1, \ldots, N-1 \quad (34)$$

Obviously, both (31) and (34) could be satisfied separately with a minimum of zero at every station in question. But in fairing we seek a compromise between the two requirements. Therefore, in the sense of least squares, we want to minimize

$$E = \sum_{n=0}^{N} \left(f(x_n) - y_n\right)^2 + s \sum_{n=1}^{N-1} \left(f''(x_n) - r_n\right)^2 \quad (35)$$

where s is a parameter we may select arbitrarily to control the severity of the fairing. The greater s, the more emphasis on the second term related to fairness, and the less weight on the first term related to the accuracy of maintaining the given offsets. From Ref. 27 we quote the experience:

s = 0 corresponds to exact interpolation, no fairing,
s = 0.1 still practically no fairing,
s = 1. conventional fairing,
s = 10. severe fairing.

In Theilheimer's method, the polynomial used in fairing is the same as the one defined in Eq. 10 for interpolation purposes. For N+1 given offsets, it contains N+3 unknown coefficients, C_n. To minimize the expression E, Eq. 35, with respect to this set of coefficients we require

$$\frac{\partial E}{\partial C_n} = 0 \;, \text{ for } n = 1, \ldots, N+3 \quad (36)$$

This results in the normal equations for the C_n.

Linear Programming Formulation of Fairing.

The term "linear programming problem" refers to a certain class of optimization problems which are of the form (Ref. 37):

Find the minimum (or maximum) of an objective function, linear in $x_1, \ldots, x_n$,

$$\sum_{i=1}^{n} c_i \cdot x_i = \text{Min. (Max.)} \qquad (37)$$

subject to the m linear constraints

$$\sum_{i=1}^{n} a_{k_i} \cdot x_i = b_k \ , \text{for } k = 1, \ldots, m \qquad (38)$$

where

x_i, optimization variables, for i = 1, ..., n, required to be non-negative ($x_i \geq 0$)

c_i, coefficients in objective function, for i = 1, ..., n

b_k, constraints, for k = 1, ..., m

m, number of constraints

n, number of variables x_i , where n > m

a_{k_i}, coefficients in constraints

There are many efficient, and readily available computer algorithms for the solution of linear programming problems, and it would therefore be of advantage to formulate the fairing problem in terms of the above format. This was done in the method developed at Todd Shipyards by Berger, Webster, Tapia, and Atkins and reported in Refs. 21, 28.

In the spirit of Eq. 34, it is required that the second derivative of the curve and the second divided difference of the given offsets have the same sign which may be expressed in the inequality

$$r_n \cdot f''(x_n) \geq 0 \ , n = 1, \ldots, N-1 \qquad (39)$$

In analogy to Eq. 31, it is also postulated that the ordinate deviation be minimized

$$| f(x_n) - y_n | \leq \lambda , \text{for } n = 0, \ldots, N \qquad (40)$$

where λ an unknown for which we seek the smallest feasible value. The variable λ constitutes the objective function of the fairing problem and is to be minimized.

Eq. 40 is equivalent to the two linear inequalities

$$\left.\begin{aligned} -\lambda + f(x_n) &\geq y_n \\ \lambda + f(x_n) &\geq y_n \end{aligned}\right\} n=0,\dots,N \qquad (41)$$

Substituting the Theilheimer polynomial, Eq. 10, and introducing new unknowns S_n, P_n, Q_n, called slack variables, we may transform Eqs. 39, and 40 into three equality constraints of the form

$$\left.\begin{aligned} &\left.\begin{aligned} -\lambda + (a + b x_n + \dots + A_{N-1}(x_n - x_{N-1})^3_{\#}) - S_n &= y_n \\ \lambda + (a + b x_n + \dots + A_{N-1}(x_n - x_{N-1})^3_{\#}) - P_n &= y_n \end{aligned}\right\} n=0,\dots,N \\ &r_n\left\{2c + \dots + 6A_{N-1}(x_n - x_{N-1})_{\#}\right\} - Q_n = 0, \; n=1,\dots,N-1 \end{aligned}\right\} (42)$$

The problem of minimizing λ subject to these constraints is now in agreement with the linear programming format of Eqs. 37 and 38. There are 3 N+1 linear constraints, Eq. 42, and 4 N+5 unknown variables, namely λ, a, b, c, A_0, ..., A_{N-1} and 3 N+1 slack variables.

It is of some importance that linear programming permits only non-negative variables, X_i, whereas in the fairing problem the coefficients in the Theilheimer polynomial may assume negative values. This problem may be overcome, at the expense of introducing still more unknowns, by expressing every conceivably negative number, A_n, as the difference of two non-negative numbers A_n', A_n''

$$A_n = A_n' - A_n''$$

The solution to the fairing problem may then be found with the aid of linear programming algorithms. For more details and for an extension of this technique to three-dimensional spline fairing, see Refs. 21, and 28.

Bakker's Spline Fairing Method.

Another interesting approach to the spline fairing problem was taken by Bakker at NSMB, Ref. 25. The fairing criterion he used aims at a compromise between the minimum offset deviation requirement, and the requirement of higher order fairness. He measures such fairness by the k'th divided differences of the faired curve (not the input points!), and has obtained the best practical results by using k = 5. He minimizes the sum of the squares of these divided differences together with the sum of the squares of the offset deviations (least squares method), applying a control parameter similar to s in Eq. 35.

1.32 Parametric Methods.

Designers have frequently argued that the offset points lifted from a lines drawing are only a coincidental intermediate result of the conventional design and lofting procedure. These lines are usually first derived from a set of geometrical concepts and conditions in the designer's mind, and if the final lines plan meets these conditions it is of no significance where any particular offset point may be located. In mathematical lines development, there is therefore no cogent need for going through the medium of offset points.

Historically, the representation of ship form or of its elements in terms of a set of shape parameters has first been undertaken with the aim of systematic variation and description of hull form families, Taylor and Weinblum, Refs. 2, 3, 8. More recently, the purpose has been extended into the design of new ship forms from a given set of parameters, Thieme and Williams, Refs. 7, 9, 23.

A system of descriptive parameters may typically consist of (see Ref. 23):

*Ship contours and regional boundaries,

for example, for any given draft:

L_{WF}, length of waterline entrance

L_{WA}, length of waterline run

L_{WMF}, length of parallel midbody, before origin

L_{WMA}, length of parallel midbody, abaft origin

B_W, maximum breadth of waterline

*Integral parameters,
for example, for any waterline under consideration, using the nondimensional coordinate system $\xi - \eta$, Fig. 8:

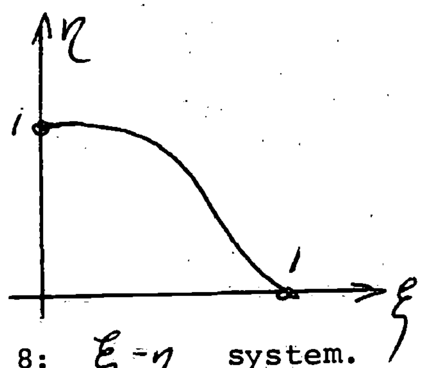


Fig. 8: $\xi - \eta$ system.

$$M_0 = \int_0^1 \eta \, d\xi$$ = waterplane area, nondim.

$$M_1 = \int_0^1 \xi \eta \, d\xi$$ = static moment, nondim.

$$M_2 = \int_0^1 \xi^2 \eta \, d\xi$$ = long. moment of inertia, nondim.

$$M_n = \int_0^n \xi^n \eta \, d\xi$$ = n'th order moment, nondim.

*Differential parameters,
for example, for a waterline as above:

$\eta'(0)$ = slope at parallel midbody

$\eta'(1)$ = slope at end

$\eta''(0)$ = curvature at parallel midbody

$\eta''(1)$ = curvature at end

$\eta(1)$ = end ordinate

The objective in design is finding functions complying with the above conditions, for example polynomials of the form

$$\eta = 1 + \sum_{i=1}^{N} a_i \xi^i \tag{43}$$

Substituting (43) into the above conditions leads to a system of linear equations for the a_i. The critical issue is selecting the appropriate number of terms in Eq. 43, and thereby the number of parameters that can be satisfied. Many terms means high orders of ξ, and hence potentially many inflection points or wiggles. Few terms may not suffice to ensure the desired character of the curve. Williams, Ref. 23, has investigated this issue, and relates his experience that the best compromise was obtained with the parameters

$$M_0, \eta'(0), \eta'(1), \eta''(0), \eta''(1), \eta(1), \eta(0) \tag{10}$$

using the terms

i = 1, 2, 3, 4, 6, 8.

The parameters need to be determined from a preliminary lines sketch, or from similar ships.

The whole design procedure, as outlined by Williams in Ref. 23, may be summarized as follows:

*Start from a preliminary lines sketch.

*Read off, and fair (graphically) the ship contours and regional boundaries. This allows to define suitable fairing intervals, and nondimensional coordinate systems. The ship ends must be either modified or treated separately because of their extreme curvatures and slopes.

*The waterline parameters are read off or computed from the drawing. They are then cross-faired (graphically) in the draft direction. This means similar sets of parameters must be used for all waterlines.

*The waterlines are now represented mathematically in accordance with their parameters. The system of equations for the a_i is sometimes ill-conditioned, particularly if some of the parameters are not fully compatible with the assumed degree of the polynomial. In any event, the appearance of the waterlines must be checked, and the design parameters may have to be modified until satisfactory lines are obtained.

*From the final form of the equations, offsets may be obtained at any point on the hull surface by interpolation.

The influence function method.

Thieme, Refs. 7, 9, has developed an interesting variant of the parametric method which permits composing a ship line from given parameters in a fast and elegant manner. In the dimensionless coordinates of Fig. 9, his approximating polynomial may be written as

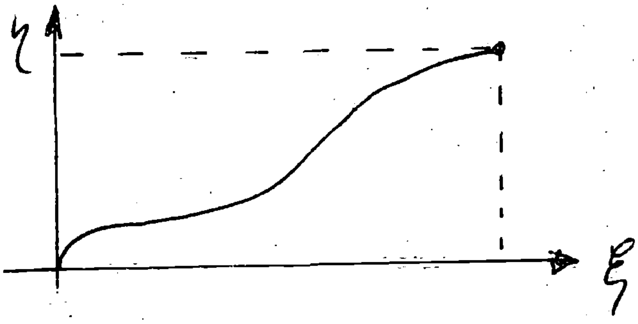


Fig. 9: Thieme's coordinate system.

$$\begin{aligned} \eta &= {\sum_{\alpha}}' C_\alpha \cdot \eta_{\alpha(\beta\gamma)} = \\ &= C_\vartheta \cdot \eta_\vartheta + C_\varepsilon \eta_\varepsilon + C_\omega \eta_{\omega\nu\sigma} + \\ &+ C_\nu \eta_{\nu\sigma} + C_\sigma \eta_\sigma + C_\mu \eta_{\mu\varrho} + \\ &+ C_\varrho \eta_\varrho + \dots \text{ (a few others, Ref. 7)} \end{aligned} \tag{44}$$

where

C_α, <u>influence coefficient</u>, constant value of the parameter α, to be given as input

η_α, singly-subscripted <u>influence function</u>, a polynomial in ξ, providing a unit value of the parameter α, and zero values for all other parameters of the set

$\eta_{\alpha\beta\gamma}$, multiply-subscripted <u>influence function</u>, a polynomial in ξ, providing a unit value of the parameter α, finite values of the parameters β, γ, etc., and zero values of all other parameters .

$C_\vartheta = \eta(1)$ = ordinate at $\xi = 1$

$C_\varepsilon = A_{WP}$ = "waterplane" area

$C_\omega = \sqrt{2\rho}$ = nose rounding at $\xi = 0$, where ρ = nondim. radius of curvature

$C_\nu = \eta'(\xi = 0,\ \omega = 0)$ = nose slope, incremental to rounding function

$C_\sigma = \eta''(\xi = 0,\ \omega = 0)$ = nose curvature, incremental to rounding and slope functions

$C_\mu = \eta'(1)$ = slope at $\xi = 1$

$C_\varsigma = \eta''(1)$ = curvature at $\xi = 1$, incremental to slope function

Ref. 7 gives details as to how the parameters C_α may be determined from a sketch or drawing. The influence functions $\eta_{\alpha(\beta\gamma)}$ are polynomials of the type of Eq. 43, and can be determined easily from the boundary conditions contained in their definition. Thieme, Ref. 7, has tabulated several basic functions and their polynomial coefficients. Fig. 10 shows two examples.

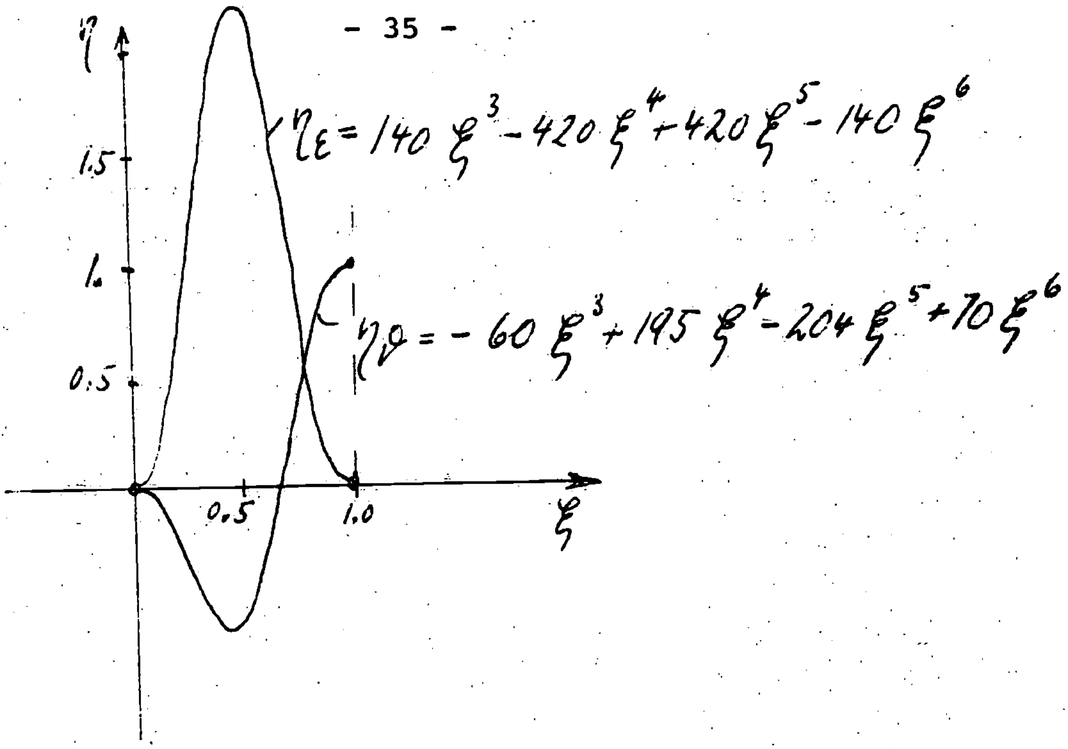


Fig. 10: Two influence functions from Ref. 7, for end ordinate, η_ϑ, and area, η_ε.

As the polynomial (44) is built up, care must be taken with coefficients such as C_ν, C_σ, C_ρ whose subscripts appear in more than one influence function. They represent incremental effects so that, for example, the slope coefficient C_ν can only be found when the slope due to the C_ω-term is known. For more procedural details see the original. As the final result, the sum of all terms in (44) may be combined into a single polynomial.

Evidently, as the number of independent parameters goes up the number of polynomial terms must increase, and the method becomes subject to the same oscillation problem as other high degree polynomial methods. But the influence function method is certainly a fast and efficient tool in composing (or decomposing) ship lines parametrically, by-passing the difficulties of solving complicated systems of equations for every new problem.

The method of Miller and Kuo.

Miller and Kuo, Ref. 20, have advanced a fairing method with some features similar to the aforementioned parametric methods.

The ship is subdivided into several regions of which certain ones such as the ends, parallel midbody, bottom and bilge, are treated separately. In the main portions of fore- and afterbody, the waterlines are approximated by polynomials of the type (43), complying with some six to ten integral and differential parameters. The coefficients of the polynomials so obtained are faired vertically in the sense of least squares.

Miller and Kuo report that they were able to represent ship surface regions by as few as 15 to 25 x-z-polynomial terms. But they do not contend they reached more than several inches accuracy, full scale, in these cases.

1.33 Direct polynomial methods.

The methods discussed in this section derive polynomial hull equations from a given set of offsets, rather than from given parameters. They proceed directly from the given hull without preparatory transformations. The example discussed is based on Refs. 15 and 18 by Kerwin et al. An analogous method by F. Taylor, using Chebycheff polynomials, is presented in Ref. 19.

It must be understood that Kerwin was aiming only at a relatively crude hull representation suitable for hydrodynamic computations. He could therefore afford to modify the waterline endings so as to obtain a rectangular contour, Fig. 11.

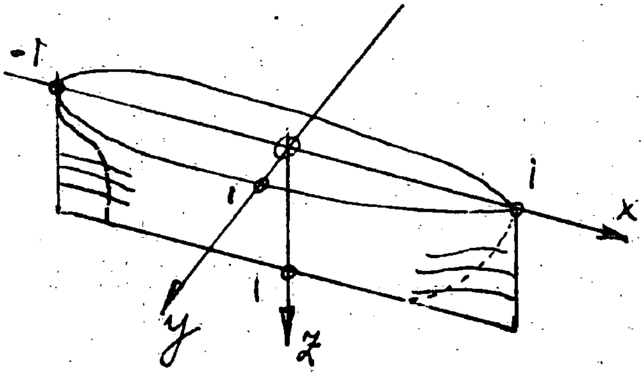


Fig. 11: Nondimensional coordinates

A suitable approximating function is then

$$g(x,z) = \sum_{n=1}^{N} \sum_{m=1}^{M} a_{mn} \psi_m \varphi_n \tag{45}$$

where

ψ_m = m'th polynomial in x

φ_n = n'th polynomial in z

The coefficients a_{mn} are to be determined from the condition that $g(x,z)$ is to approximate the given offsets f_{ij} in the sense of least squares.

$$E = \sum_{j=1}^{J} \sum_{i=1}^{I} \left\{ f_{ij} - \sum_{n=1}^{N} \sum_{m=1}^{M} a_{mn} \psi_{mj} \varphi_{ni} \right\}^2 \tag{46}$$

where I, J = number of given z-, and x-location of offsets

Generally, around 100 coefficients a_{mn} will be required, and an attempt to find these unknowns simultaneously from the system of equations corresponding to (46) does not promise any success, Ref. 15.

Instead, Kerwin first approximates the stations, then their coefficients in the x-direction.

For the j'th station: $$\bar{g}_j(z) = \sum_{n=1}^{N} \bar{a}_{nj} \varphi_n \tag{47}$$

for example, as follows (dropping j):

$$\bar{g}(z) = \bar{a}_1 + \bar{a}_2 z + \bar{a}_3 z^2 + \bar{a}_4 z^3 + \bar{a}_5 z^{100} + \bar{a}_6 z^{200} \tag{48}$$

The higher order terms are appropriate for flat bottom and bilge area description.

Solving for the coefficients $\bar{a}_n$ by applying the least square criterion directly, may again lead to numerical trouble due to ill-conditioned systems of equations. This is why orthogonal polynomials were introduced by Kerwin.

Before giving more details, a few comments about the advantages of orthogonalization shall be made. The student familiar with the subject may skip the following several paragraphs.

Suppose a function $f(x)$ is given by N offsets at the stations x_j: $f_1, \ldots, f_j, \ldots, f_N$

The function shall be approximated by the function

$$F(x) = \sum_{k=1}^{K} a_k P_k(x) \quad , \quad K \leq N \tag{49}$$

where $P_k(x)$ = k'th function of an orthogonal set, see Eq. 52

The least squares criterion requires

$$E = \sum_{j=1}^{N} \Big(f_j - \sum_{k=1}^{K} a_k P_k(x_j) \Big)^2 = Min. \tag{50}$$

or

$$\frac{\partial E}{\partial a_m} = 0 = 2 \sum_{j=1}^{N} \Big(f_j - \sum_{k=1}^{K} a_k P_k(x_j) \Big) P_m(x_j) \tag{51}$$
$$m = 1, \ldots, K$$

The orthogonality property for the function set is expressed as

$$\sum_{j=1}^{N} P_k(x_j) \cdot P_m(x_j) = \begin{cases} 0 \, , & \text{for } m \neq k \\ \sum_{j=1}^{N} P_m^{\,2}(x_j) \, , & \text{for } m = k \end{cases} \tag{52}$$

From (51) and (52)

$$\sum_{j=1}^{N} f_j P_m(x_j) = \sum_{j=1}^{N} a_m P_m^2(x_j) \tag{53}$$

$$a_m = \frac{\sum_{j=1}^{N} f_j P_m(x_j)}{\sum_{j=1}^{N} P_m^2(x_j)}, \quad m=1,\dots,K \tag{54}$$

or, in matrix form,

$$\begin{bmatrix} \sum_j P_1^2(x_j) & 0 & 0 \dots & 0 \dots & 0 \\ 0 & \sum_j P_2^2(x_j) & 0 \dots & 0 \dots & 0 \\ 0 & 0 & \sum_j P_3^2(x_j) \dots & 0 & \dots 0 \\ - & - & - & - & - \\ 0 & 0 & 0 \dots & \sum_j P_m^2(x_j) & 0 \\ - & - & - & - & - \\ 0 & 0 & 0 \dots & 0 \dots & \sum_j P_K^2(x_j) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \dots \\ a_m \\ \dots \\ a_K \end{bmatrix} = \begin{bmatrix} \sum_j f_j P_1(x_j) \\ \sum_j f_j P_2(x_j) \\ \sum_j f_j P_3(x_j) \\ \dots \\ \sum_j f_j P_m(x_j) \\ \dots \\ \sum_j f_j P_K(x_j) \end{bmatrix} \tag{55}$$

In summary, for orthogonal functions the system of normal equations reduces to one independent equation for each coefficient a_m, of the form (54). If on the other hand the orthogonality relation (52) is not satisfied, the elements in the left-hand matrix of (55) will be finite everywhere, rather than just in the main diagonal. In this case the accuracy losses involved in equation solving or matrix inversion cannot be avoided.

Kerwin uses a set of polynomials that are only approximately orthogonal in two respects:

First, he uses Legendre polynomials, modified to be orthogonal in the interval $z = 0...1$, with respect to integration rather than summation as in Eq. 52. These functions satisfy the relation

$$\int_0^1 \varphi_k(z)\cdot\varphi_m(z)\,dz = \begin{cases} 0, \text{ for } m \neq k \\ \int_0^1 \varphi_m^2(z)\,dz, \text{ for } m = k \end{cases} \qquad (56)$$

but summation as in (52) yields only an approximate zero for $m \neq k$.

The modified Legendre functions are

$$\begin{aligned} \varphi_1 &= 1 \\ \varphi_2 &= 2z-1 \\ &- - - - - - - \\ \varphi_{n+1} &= \frac{2n+1}{n+1}(2z-1)\varphi_n - \frac{n}{n+1}\varphi_{n-1} \end{aligned} \qquad (57)$$

Second, Kerwin also includes high order polynomial terms like z^{100}, z^{200} in his function set since they are at least approximately orthogonal and need not be replaced by Legendre functions.

In summary, his approximating function, corresponding to the example of Eq. 48, is in matrix notation

$$\bar{g}(z) = [\bar{a}']\cdot[\varphi'] \qquad (58)$$

where

$$[\bar{a}'] = [\bar{a}_1' \;\ldots\ldots\; \bar{a}_6'] \;,\; [\varphi'] = \begin{bmatrix} \varphi_1' \\ \vdots \\ \vdots \\ \varphi_6' \end{bmatrix} = \begin{bmatrix} 1 \\ 2z-1 \\ \vdots \\ z^{100} \\ z^{200} \end{bmatrix}$$

The coefficients, $\bar{a}_n'$, are determined from a least squares condition analogous to (51). This leads to a system of equations analogous to (55), but due to the approximate degree of orthogonality the elements off the main diagonal in the first matrix of (55) will not completely vanish. They will be relatively small, however, and the predominance of the main diagonal will ensure a healthy numerical solution of the system of equations.

As indicated earlier, the coefficients $\bar{a}_n'$ obtained at the given stations are subsequently approximated by a longitudinal least squares fit, in a manner very similar to the station fit. For the n'th polynomial coefficient

$$\bar{a}_n'(x) = \sum_{m=1}^{M} a_{mn}' \, \psi_m' \tag{59}$$

where ψ_m' = orthogonal polynomial in x.

When the a_{mn}' are found, $g(x,z)$, Eq. 45, can be written as a polynomial in powers of x and z, substituting results from (47) and (59).

Kerwin's method of orthogonal polynomial fitting drastically simplifies the task of finding many hull polynomial coefficients accurately. But it cannot avoid the difficulties inherent in the attempt to find a single polynomial hull equation for the whole ship. As a result, the method is suitable whenever an equation is sought that represents, in a somewhat approximate sense, a realistic shiplike shape.

1.34 Transformation methods.

The methods to be discussed under this heading use basically the same fitting and fairing techniques as those of the preceding sections, but these techniques are applied to certain transformed properties of the ship instead of the hull surface given originally. The advantage is sought in simplifications and improved quality of the fairing process proper.

Pien's method.

P. C. Pien, Ref. 13, has a three-step procedure;

*The shape y = f(x,y) , which is given by its offsets, is transformed into a simplified form, $\bar{f}(x,z)$, with vertical stem and stern contours:

$$\bar{f}(x,z) = \frac{f(x,z)}{\left\{1-\left(\frac{x}{P(z)}\right)^2\right\}^{1/M}} \qquad (60)$$

where P(z) = equation of the respective contour,

$$0 \leqslant X < P(z)$$

M = an arbitrary, relatively great constant, say, 10.

The location, x_{max}, of the new stem or stern is arbitrary provided that $x_{max} < P(z)$.

*The modified hull surface is then interpolated, meeting the conditions at three given stations exactly:

$$\bar{f}_1(x,z) = \sum_{i=1}^{3} \bar{f}(x_i,z) \cdot P_i(x) \qquad (61)$$

where

x_i, given stations for interpolation, for example midship section, an intermediate point, and the end point x_{max}.

$\bar{f}(x_i,z)$, equation of section at x_i, obtained by two-dimensional fitting.

$P_i(x)$, interpolating polynomials, determined from the conditions

$$P_i(x_j) = \begin{cases} 1, & \text{for } i=j \\ 0, & \text{for } i \neq j \end{cases}$$

*Between the stations used in (61), there remains a finite error, which according to Pien's experience is relatively easy to fit by a least squares polynomial:

$$\bar{f}_2(x,z) = \bar{f}(x,z) - \bar{f}_1(x,z) \approx \sum_i \sum_j a_{ij} x^i z^j \tag{62}$$

If one assembles the results of the three steps, he obtains the hull equation

$$y(x,z) = \left\{\bar{f}_1(x,z) + \sum_i \sum_j a_{ij} x^i z^j\right\}\left[1-\left(\frac{x}{P(z)}\right)^2\right]^{1/M} \tag{63}$$

Note that function vanishes at stem and stern, x = P(z).

The method of Rösingh and Berghuis.

In the method of Ref. 12 by Rösingh and Berghuis, fairing is applied to a function closely related to the second derivative of the waterline. The waterline is then obtained by double integration. Thus, the method constitutes a genuine transformation method.

In the dimensionless coordinates of Fig. 12, a waterline may be represented as

$$\eta(\xi) = 1 - \xi^3\left[1 + a_1(1-\xi) - \int_1^{\xi}\int_1^{\bar{\bar{\xi}}} \kappa(\bar{\xi})\, d\bar{\xi}\, d\bar{\bar{\xi}}\right] \tag{64}$$

where $a_1 = \eta'(1)$ = end slope, hence for $\kappa(\xi)$:

$$\kappa(\xi) = \frac{d^2}{d\xi^2}\left\{\frac{\eta(\xi) - 1 + \xi^3}{\xi^3}\right\} \tag{65}$$

Since

$$\int_1^{\xi}\int_1^{\bar{\bar{\xi}}} \kappa(\xi)\, d\bar{\xi}\, d\bar{\bar{\xi}} = \frac{\eta(\xi) - 1 + \xi^3}{\xi^3} + a_1(1-\xi)$$

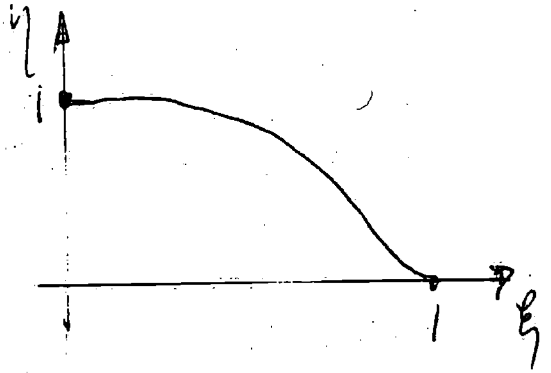


Fig. 12: Waterline coordinates

The function $\kappa(\xi)$ is a somewhat modified curvature. The desired values for κ are read off the drawing, using finite difference approximations for the derivatives. Then $\kappa(\xi)$ is approximated by some suitable function in the sense of least squares. The approximating function is a smooth function, having continuous first, and second derivatives. The waterline, obtained by double integration from (64), is therefore guaranteed to be smooth also, and the integration will further filter out minor oscillations. Eq. 64 also accounts for some of the essential boundary conditions of the waterline, namely ordinates and slopes at $\xi = 0$, and 1. Vertical fairness is accomplished by a separate crossfairing step like in other draft function methods.

The feature that the method exploits the smoothing qualities of integration, is its essential advantage. For example, the function $\kappa(\xi)$ may be approximated rather crudely without too much distortion of the desired lines character. This simplifies the fitting problem a great deal. On the other hand, the lines cannot easily be kept close to a given set of points.

The method of von Kerczek and Tuck.

Ref. 29 describes a method by Tuck and von Kerczek which was originally motivated by the intention to compute hydrodynamic quantities, such as streamlines and pressure distributions, for a given ship hull on the basis of slender body theory. An elegant and flexible hull form representation, at least for the underwater hull, came out as a by-product.

In this method, the sections are generated by conformal mapping, and the coefficients of the mapping function are faired longitudinally.

The mapping function for a section has the form

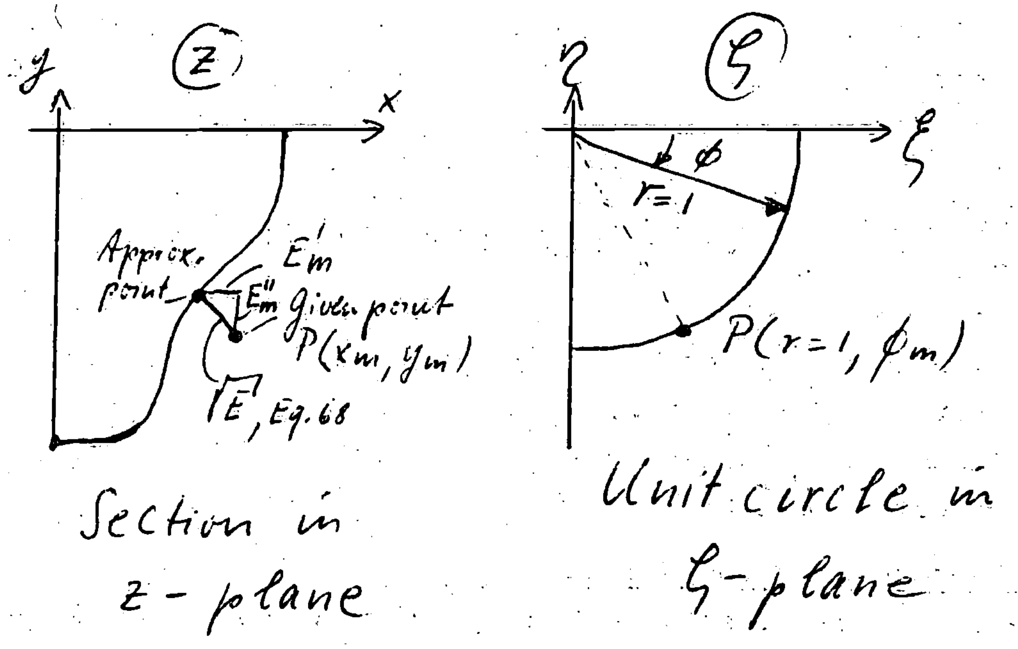
$$z = \sum_{n=1}^{N} a_n \cdot \zeta^{3-2n} \tag{66}$$

where

$z = x + iy$, x, y = coordinates of section, Fig. 13

$\zeta = \xi + i\eta = re^{i\phi}$, r = 1, coordinates of unit circle, Fig. 13

The real coefficients, a_n, are determined so that the least square error between the curve (66) and the given offset points is minimized.



Section in z-plane

Unit circle in ζ-plane

Fig. 13: Conformal mapping, coordinates

For a given point (x_m, y_m), with (66),

$$x_m = \sum_{n=1}^{N} a_n \cos((3-2n)\phi_m) + E_m' \tag{67}$$

$$y_m = \sum_{n=1}^{N} a_n \cdot \sin((3-2n)\phi_m) + E_m''$$

where E_m', E_m'' = error components.

Hence $$E = \sum_{m=1}^{M} \left(E_m'^2 + E_m''^2\right) = Min. \tag{68}$$

M = number of given points,

or $$\frac{\partial E}{\partial a_n} = 0 \, , \quad n = 1, \ldots, N$$

$$\tag{69}$$

and $$\frac{\partial E}{\partial \phi_m} = 0 \, , \quad m = 1, \ldots, M$$

For more details see the original reference. The longitudinal least squares fit of the a_n is based on the polynomial

$$a_n(S) = \sum_{p=1}^{P} \alpha_{pn} S^{p-1} \tag{70}$$

S, longitudinal coordinate.

The nature of conformal mapping ensures smooth character of the final result. Note that in conformal mapping one obtains a horizontal tangent at the keel, and a vertical tangent in the waterline as a consequence of the conformity of angles between ζ-plane circle and z-plane section. In practice, flare may be provided up to a region right under the waterline.

The method may be extended to include the above water portion of the ship by moving the mapping planes y = 0, η = 0 far enough upward. In this event, the hydrodynamic significance of the result is sacrificed.

Further extensions of the method encompassing parametric design constraints are under development at The University of Michigan by Ogilvie and Bhattacharyya.

1.35 Surface fitting methods.

The methods discussed up to this point were either draft function or sectional methods, in which the three-dimensional fairing problem is reduced to a sequence of two-dimensional ones. For example, the waterlines were faired first, then the sections, and so forth by trial and error. This approach to hull fairing may have been motivated by the natural inclination of the naval architect to simulate the spline, a two-dimensional fairing tool. More importantly, it was believed that any attempt to obtain a surface equation in a single fairing step would inevitably lead to a prohibitive increase of the equation solving effort. It turns out that both of these opinions are not the final verdict.

In other industries, for example in automobile and aircraft production, equally complex three-dimensional shapes have to be dealth with. But these industries have never been exclusively attracted by two-dimensional layout methods such as by spline. Instead, the clay model and the three-dimensional mock-up governed the thinking, and not so surprisingly these became the ancestors of three-dimensional mathematical surface representation and fairing.

The major obstacle of having to determine too many unknowns in a single step may be overcome by working in suitable surface coordinate systems, and/or by complete orthogonalization.

Coons' Surfaces.

S. A. Coons, Ref. 40, has formulated a parametric surface representation which has found wide application in the aircraft and automobile industries, and particularly in computer graphics applications (see for example the sections by Herzog and Chasen in Ref. 36). It has also been extended to ship surfaces by Hamilton and Weiss, Ref. 24. Brief introductory descriptions of the method are contained in Refs. 24, and 36 (Chasen).

Coons represents the surface in terms of two parameters (coordinates) u, and v. The equation of a quadrilateral surface patch, Fig. 14, is given as

$$\begin{aligned}\bar{r}(u,v) = {} & \bar{r}(0,v)\cdot F_0(u) + \bar{r}(1,v)\cdot F_1(u) + \\ & + \bar{r}(u,0)\cdot F_0(v) + \bar{r}(u,1)\cdot F_1(v) - \\ & - \bar{r}(0,0)\cdot F_0(u)\cdot F_0(v) - \bar{r}(0,1)\cdot F_0(u)\cdot F_1(v) \qquad (71) \\ & - r(1,0)\cdot F_1(u) \bullet F_0(v) - \bar{r}(1,1)\cdot F_1(u)\, F_1(v)\end{aligned}$$

where $\bar{r}$ denotes either x, or y, or z so that (71) contains the parameter equations

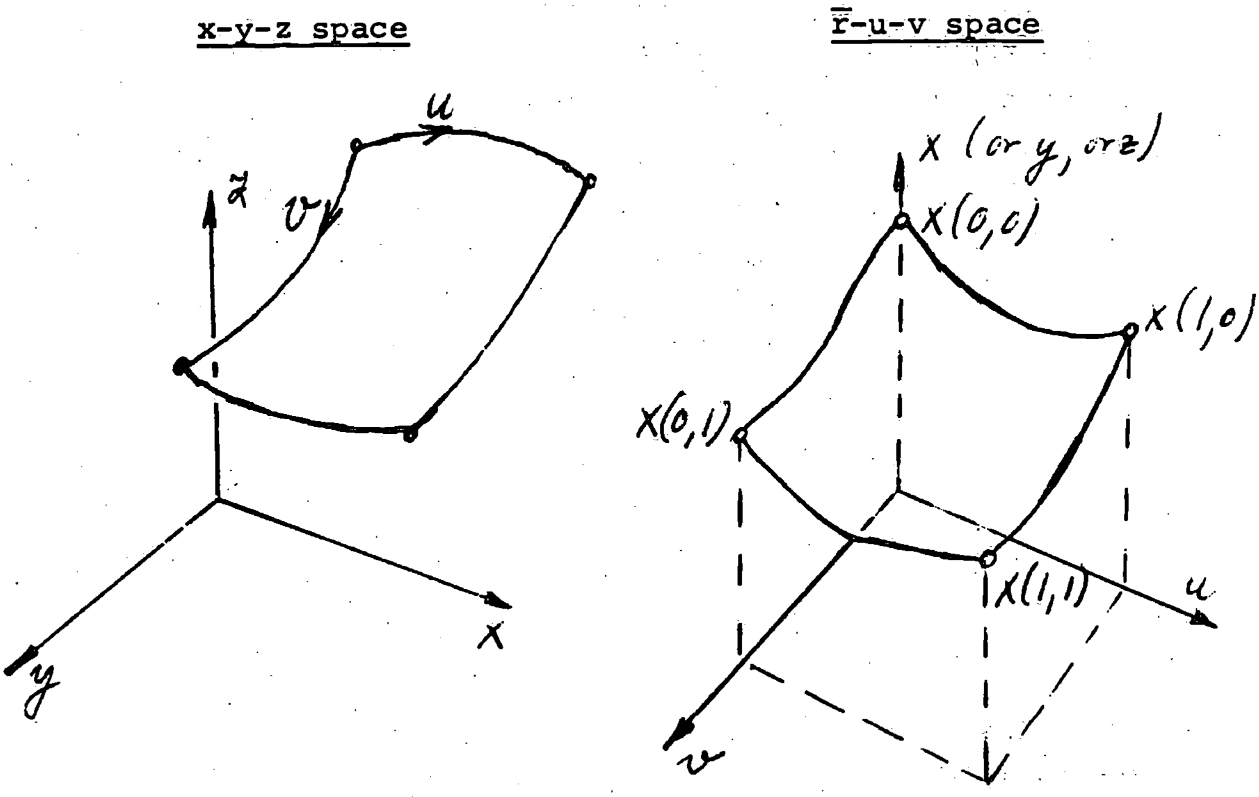
$$x = x(u,v)$$
$$y = y(u,v)$$
$$z = z(u,v)$$

$\bar{r}(0,v)$, $\bar{r}(1,v)$, $\bar{r}(u,0)$, $\bar{r}(u,1)$ are the parameter equations of the four patch boundaries.

$\bar{r}(0,0)$, $\bar{r}(0,1)$, $\bar{r}(1,0)$, $\bar{r}(1,1)$ are the coordinates of the patch corner points.

$F_0(u)$, $F_0(v)$, $F_1(u)$, $F_1(v)$ are blending functions defined for the purpose of interpolation within the patch, see below.

The coordinates u, v represent curvilinear surface coordinates for the patch in x-y-z space. They are defined so that the corners of the patch in x-y-z space and in $\bar{r}$-u-v space correspond to each other. But the exact relation between u, v and the components of $\bar{r}$ depends on the shape of the patch as expressed in Eq. 71.



x-y-z space

$\bar{r}$-u-v space

Fig. 14: Coons' coordinates

In the interest of convenient surface manipulation and matching of contiguous patches, it is desirable to define the blending functions so that the patch boundary curves depend only on the coordinates and slopes at the patch corners, rather than also on slopes and points along the boundaries or within the surface.

In summary, the blending functions shall satisfy the following conditions:

*The boundary curves must lie within the surface. This is accomplished if

$$F_0(0) = 1 \qquad F_1(0) = 0 \qquad (72)$$
$$F_0(1) = 0 \qquad F_1(1) = 1$$

where the arguments may be u, or v.

*The slopes along each boundary shall depend only on the corner point slopes of that boundary.

Consider, for example, the general expression for the u-direction slope along the boundary u = 0:

$$\begin{aligned}\frac{\partial \bar{r}(0,v)}{\partial u} = {} & \bar{r}(0,v)\cdot\frac{\partial F_0(u=0)}{\partial u} + \bar{r}(1,v)\cdot\frac{\partial F_1(u=0)}{\partial u} + \\ & + \frac{\partial \bar{r}(0,0)}{\partial u} F_0(v) + \frac{\partial \bar{r}(0,1)}{\partial u} F_1(v) - \\ & - \bar{r}(0,0)\frac{\partial F_0(u=0)}{\partial u} F_0(v) - \bar{r}(0,1)\frac{\partial F_0(u=0)}{\partial u}\cdot F_1(v) - \\ & - \bar{r}(1,0)\frac{\partial F_1(u=0)}{\partial u} F_0(v) - \bar{r}(1,1)\frac{\partial F_1(u=0)}{\partial u} F_1(v)\end{aligned} \qquad (73)$$

This expression would depend on the properties of other boundaries unless we let

$$\frac{\partial F_1(u=0)}{\partial u} = 0$$

By similar argument for other slopes and other boundaries

$$F_0'(0) = 0,\ F_1'(0) = 0,\ F_0'(1) = 0,\ F_1'(1) = 0, \qquad (74)$$

where the prime denotes the derivative by u for $F_0(u)$, $F_1(u)$, and correspondingly for v.

Eqs. 72 and 74 furnish four conditions for every F. These conditions are satisfied by the cubics

$$\begin{aligned} F_0(u) &= 1 - 3u^2 + 2u^3 \\ F_1(u) &= 3u^2 - 2u^3 \end{aligned} \qquad (75)$$

and identical forms for $F_0(v)$, and $F_1(v)$. Note that $F_0 + F_1 = 1$, as one would expect from coordinate invariance.

In conclusion, the cubics (75) are suitable to represent a patch in terms of its boundary curves, to vary any boundary without alteration of the opposite boundary, and to match boundaries of contiguous patches.

In principle, blending functions could be constructed to meet more conditions at the expense of some added complexity. Chasen, Ref. 36, mentions the use of an additional parameter K which is adjusted so as to ensure agreement between surface equation and given offset at the center point of the patch. This leads to blending functions such as

$$F_0(u) = (16K - 10)u^4 + (21 - 32K)u^3 + (16K - 12)u^2 + 1 \qquad (76)$$

The representation of the boundaries, $\bar{r}(o,v)$, etc., is arbitrary in principle. But if one wants to prescribe corner point offsets and slopes, and keep the number of internal inflection points limited, cubics suggest themselves for this purpose also, for example:

$$\bar{r}(o,v) = \sum_{i=0}^{3} a_i v^i \qquad (77)$$

Substituting (75) and (77) into (71), one can see that one obtains a bicubic equation for $\bar{r}(u,v)$:

$$\bar{r}(u,v) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} u^i v^j \qquad (78)$$

In practice, for any patch of given corner point offsets and slopes, one will first derive the boundary equations (77), and then, using the universal functions (75), obtain the bicubic (78) by assembling terms in (71).

Derived properties of the surface, like area, centroid, moments, etc., can be suitably expressed in terms of the u-v coordinates, see Ref. 24. It is also relatively simple to get oriented views which explains the popularity of Coons' surfaces in computer graphics.

Coons' surface representation will accomplish an exact interpolation of a given set of points and slopes if these points can be used as patch corner points. There is no immediate control over the fairness of the surface, and bad data points may cause bumps. There has been some experimentation in active computer graphics to use light pen instructions to correct the patch corner properties until the shape is satisfactory. It is also conceivable to combine this surface representation method with some fairing technique, and to incorporate design constraints.

Lidbro's method.

Lidbro, Ref. 16, describes a method aiming at the interpolation of a given surface by means of patches like the Coons method, but less ambitious mathematically. The steps involved are:

*The surface is subdivided into many patches. For ships, the sections may be chosen as one set of boundaries (called generating curves), whereas the second set is formed by curved lines in the longitudinal direction (called parameter curves) which are neither waterlines nor buttocks, but something in between, not unlike shell plating seams. These must be selected with some judgment.

- *The y- and z-offsets of the parameter curves are now approximated separately by suitable functions in x, not necessarily polynomials, see Ref. 16.

$$\left.\begin{array}{l} y = f_1(x) \\ z = f_2(x) \end{array}\right. \quad \text{for each long. boundary}$$

- *The parameter curves define points on the sections through which an interpolating function, $y = f(z)$, is now laid at the patch boundaries, and perhaps at some intermediate stations. Again, any suitable function will be used for the purpose, mainly polynomials and arcs of conic sections. The coefficients of these functions are then represented as a function of x.
- *The surface is thus uniquely defined. Hull offsets y may be interpolated for any given x, and z.

The method does not do any fairing. The accuracy of interpolation depends on the patch size and proper choice of approximating function. The results may be as good as desired, but experienced judgment is required.

The method of Kaiser, Nagel, and Schiller.

An elegant and successful surface fitting method, similar to, but in some respects more advanced than the basic Coons method, was presented by the above-mentioned authors in Ref. 41. Some of the main features are:

- *Smoothing of the input points precedes the interpolation. More details on this will be given in section 1.4.
- *A system of grid lines similar to the Coons patch boundaries is introduced. Arc lengths along the grid lines, or a close approximation thereof, are used as parametric surface coordinates. In this representation the surface may assume any orientation in space.
- *The equations of the boundaries are assumed as fifth degree polynomials in the arc length coordinate, u, for each of the dependent variables x, y, and z:

$$\left.\begin{matrix} x \\ y \\ z \end{matrix}\right\} = \sum_{k=0}^{5} A_k u^k , \text{ where } A_k = \left\{\begin{matrix} a_k \\ b_k \\ c_k \end{matrix}\right. \qquad (79)$$

This allows offsets, slopes, and curvatures to be specified and matched at both ends of the curve.

*The offsets are read from the drawing, but slopes and curvatures are treated as unknowns initially except perhaps at the very ends. The slopes and curvatures at intermediate data points are determined from the extra condition that the third and fourth derivatives shall be continuous at these points. For N - 2 intermediate points, a system of equations for 2(N - 2) unknowns will furnish all unknown slopes and curvatures. These results, together with the given points, define the polynomials (79) for every segment. The whole procedure ensures <u>reasonable fairness</u>.

*Further offsets may be derived by interpolation in the surface coordinate system. Other properties can also be derived as in the Coons method. But the computational effort should be a little greater due to the higher degree of polynomial.

<u>The method of Kantorowitz.</u>

A recent method that has already found wide application in shipbuilding was presented by Kantorowitz in Ref. 31. In its essential part, it approximates the surface in Cartesian coordinates, but taking advantage of complete orthogonality of the approximating polynomials. The major steps are as follows:

*The surface is subdivided along lines where curvature or slope discontinuities or other abrupt changes are desired. Many of the patches and their boundary curves so obtained can be approximated by two-dimensional polynomials and arcs of conics in a conventional way.

*Three-dimensional fairing is applied to the curved surfaces of forebody and afterbody between the parallel middlebody and stem or stern. Sometimes these surfaces are further subdivided. The subdivision is made so that extreme slopes and curvature transition are avoided within the surface.

The surface equation is basically a polynomial of the form

$$P(x,z) = \sum_{i=0}^{I} \sum_{j=0}^{J} a_{ij} x^i z^j \qquad (80)$$

where Q = (I + 1)·(J + 1) = number of polynomial terms, typically around 50.

The surface may also be expressed in terms of the function set $L_t(x,z)$

$$P(x,z) = \sum_{t=1}^{Q} a_t L_t(x,z) \qquad (81)$$

where $L_t(x,z)$ = orthogonal polynomial satisfying the relation:

$$\sum_{r=1}^{R} w_r L_u(x_r,z_r) \cdot L_v(x_r,z_r) = \begin{cases} 0, \text{for } u \neq v \\ 1, \text{for } u = v \end{cases} \qquad (82)$$

R = number of given points

x_r, y_r, z_r = given offsets of r'th point.

w_r = a weighting factor; great weights are used for points on the boundary which are to remain fixed. See Eq. 83.

The process of generating the polynomials $L_t(x,z)$ for any given set of data points is described briefly in Ref. 31, and some of the fundamentals are found in Refs. 42, 43.

The coefficients a_t are determined one after another from the least square condition

$$\sum_{r=1}^{R} w_r \left(P(x_r,z_r) - y_r\right)^2 \qquad (83)$$

The method may claim it has carried the principles of patch subdivision and orthogonalization nearly to the optimum. This results in an efficient and accurate procedure. The full scale fairing deviations are reported to be nowhere more than 2 inches. The input data must have been carefully checked for inconsistencies (by computer).

1.4 Data smoothing.

Most hull representation methods are sensitive to bad input data points. The fundamental difficulty lies in distinguishing input errors due to misreading and mistranscription from points that are inconsistent only because of lack of fairness in the raw data. The former are random and must be eliminated, the latter somehow express the designer's intentions and their character is to be conserved. Smoothing is here defined as the elimination of random input errors, see also page 6.

The safest, and perhaps the only foolproof, way of checking input data for random errors is graphical display and inspection of the input. This is a common procedure since graphical devices are usually available where hull fairing is done.

Analytic smoothing methods have been used with some success also, Refs. 21, 28, 31, 41, but are of limited generality. They test for closely spaced inflection points, irregularities in the higher order differences, or any other known simple property of ship data. These criteria help to detect one bad data point among several good ones, but are not reliable when more than one mistake is involved. Punching mistakes can be detected with the aid of simple check sums of several input values prepared manually and read in along with the data.

1.5 Comparison of hull representation techniques.

A general overview of hull representation techniques is given in Table I on pages 10 to 12. In that table, the classification by purpose, input, procedure, function type, etc. follows the general categories listed in pages 9 and 13. Note that the column "Section" refers to the five basic methods distinguished in page 13 and discussed in sections 1.31 through 1.35. In the column "Patches" the entry "2" means that forebody and afterbody are the only separate regions. To summarize the essential properties of the five basic methods, we give the following condensed comparison.

Spline methods.

Accuracy (Suitability for exact fit): Excellent, at least for the more advanced versions of the method.

Flexibility (Suitability for creation of lines): Good, although constraints need to iterated for.

Effort: For N+1 given data points, N+3 simultaneous equations must be solved both for spline interpolation and least square spline fairing. The linear programming formulation of the spline fairing problem leads to 4N+5 unknowns in the same case. The systems of equations are relatively well-conditioned, in general.

Quality (Smoothness): Excellent.

Parametric methods.

Accuracy: Point accuracy is not the purpose.

Flexibility: Good; immediate relation to constraints, but limited number of constraints.

Effort: Number of equations equals number of constraints; no more than 6 to 10 per line can be afforded without risking wiggles.

Quality: Only good if number of constraints relatively small.

Direct polynomial methods.

Accuracy: Limited where abrupt curvature changes are involved.

Flexibility: Not too good due to complicated relationship to constraints.

Effort: In orthogonal form, the effort is reasonable. It corresponds to the number of points per line (N), hence is about the same as for the spline method.

Quality: A tendency to show wiggles as the number of terms increases.

Transformation methods.

Accuracy: Good provided that sufficient patch subdivision is used.

Flexibility: Acceptable; iteration needed to meet constraints.

Effort: Relatively high; transformation effort is added to that of the aforementioned methods.

Quality: Good in some transformation methods.

Surface fitting methods.

Accuracy: Excellent.

Flexibility: Good; but iteration needed to meet constraints.

Effort: Relatively high; must find parameter representation of surface and solve the fitting problem.

Quality: Good.

Discussion of some special features.

*Type of function:

Discontinuous polynomials (Theilheimer), elliptic functions (Gospodnetić), variable spline polynomials (Söding), orthogonal polynomials (Kantorowitz and others) have all shown good suitability in fairing whereas straightforward polynomials must be limited to simpler cases.

*Patches:

Sufficient patch subdivision seems to be a necessary feature of any successful method. In the surface techniques and many others, the patch boundaries are directly prescribed, in the spline methods analogous transitions with discontinuous curve properties are permitted at any given data point. At the least, patch boundaries are required to accommodate abrupt change in the surface derivatives. At the maximum level of subdivision, one will select enough patches to allow patch description by simple functions.

*Draft function versus sectional versus surface fitting methods:

Both, draft function and iterative sectional methods appear somewhat less elegant than a simultaneous fit of the surface, and after the introduction of orthogonalized surface representations the computational effort does not seem to make a great difference among the methods any longer.

*Future outlook:

The problems of hull form interpolation and fairing appear to be sufficiently under control now, only some efficiency remaining to be gained. But no fully satisfactory method of lines creation seems to exist. More foolproof methods, more directly related to the constraints, will be required in the future.

1.6 References 1 through 44.

1. Chapman, F. H., A Treatise on Ship-Building, 1760, English translation, Cambridge and London, 1820.

2. Taylor, D. W., "Ship Calculations, Resistance and Propulsion," Int. Congress of Engineering, San Francisco, 1915.

3. Weinblum, G., "Exact Waterlines and Sectional Area Curves," in German, Schiffbau, 1934.

4. Benson, F. W., "Mathematical Ships' Lines," I.N.A., 1940.

5. Watanabe, K., "Mathematical Expression of Ship Forms," Journal of Zosen Kiokai, Vol. 77, 1946.

6. Lackenby, H., "On the Systematic Geometrical Variation of Ship Forms," I.N.A., 1950.

7. Thieme, H., "Systematic Development of Ship Lines," in German, Schiff und Hafen, 1952.

8. Weinblum, G., "Systematic Development of Hull Forms," in German, S.T.G., 1953.

9. Thieme, H., "On Fundamentals for the Mathematical Lines Plan of a Cargo Ship," in German, Schiffstechnik, 1956.

10. Taggart, R., "Mathematical Fairing of Ship Lines for Mold Loft Layout," A.S.N.E., 1955.

11. Theilheimer, F., "Univac Calculations of Ship Lines," Symposium on the Solution of Engineering Problems at the Applied Mathematics Laboratory, Bu Ships, 1956, and excerpts in A.S.N.E., May 1957.

12. Rösingh, W., and Berghuis, J., "Mathematical Shipform," I.S.P., 1959.

13. Pien, P. C., "Mathematical Ship Surface," I.S.P., 1960.

14. Martin, D. W., "Polynomial Representation of Some Ship Section Area Curves and the Calculation of the Associated Wave Resistance," Quart. Journ. Mech. and Applied Math., Vol. 14, 1961.

15. Kerwin, J. E., "Polynomial Surface Representation of Arbitrary Ship Forms," J.S.R., 1960.

16. Lidbro, N., "Analytic Form Definition of Ships," in German, Schiffstechnik, 1961.

17. Theilheimer, F., Starkweather, W., "The Fairing of Ship Lines on a High-Speed Electronic Computer," D.T.M.B. Report 1474, 1961.

18. Gerritsma, J., Kerwin, J. E., Newman, J. N., "Polynomial Representation and Damping of Series 60 Hull Forms," I.S.P., 1962.

19. Taylor, F., "Computer Applications to Shipbuilding," I.N.A., 1962.

20. Miller, N. S., Kuo, C., "The Mathematical Fairing of Ship Lines," European Shipbuilding, 1963.

21. Berger, S. A., Webster, W. C., "An Application of Linear Programming to the Fairing of Ships' Lines," Recent Advances in Mathematical Programming, McGraw-Hill, New York, 1963.

22. Poutous, P., "The Mathematical Design of Shiplines," N.E.C.I., 1964.

23. Williams, A., "Mathematical Representation or Ordinary Ship Forms," Schiff und Hafen, 1964.

24. Hamilton, M. A., Weiss, A. P., "An Approach to Computer-Aided Preliminary Design," MIT M. Sc. Thesis, 1964.

25. Bakker, A. R., "Application of a Computer to Some Shipbuilding Problems," I.S.P., 1965.

26. Gospodnetić, D., "Numerical Definition of Ships' Hulls by Means of Elastic Interpolation," National Research Council of Canada, Mech. Eng. Report MB-258, Ottawa, 1965.

27. Corin, T., "Recent Developments in Ship Lines Fairing at the David Taylor Model Basin," Symposium on Computer-Aided Ship Design, Washington, August 1966.

28. Berger, S. A., Webster, W. C., Tapia, R. A., Atkins, D. A., "Mathematical Ship Lofting," J.S.R., 1966.

29. Tuck, E. O., von Kerczek, C., "Streamlines and Pressure Distributions on Arbitrary Ship Hulls at Zero Froude Number," J.S.R., 1968.

30. Söding, H., "The Mathematical Treatment of Graphically Designed Hull Forms," in German, Ph.D. Thesis, Technical University of Hanover, Germany, 1966.

31. Kantorowitz, E., "Mathematical Definition of Ship Surfaces," Danish Ship Research Institute, Report No. DSF-14, Lyngby, 1967.

32. Saunders, H. E., Hydrodynamics in Ship Design, Vol. 2, Chapter 49, S.N.A.M.E., 1957.

33. Hershey, A. V., "Derivation of Mathematical Ship Shapes from the Taylor Standard Series," U.S. Naval Weapons Laboratory, Report 1737, 1961.

34. Birkhoff, G., Garabedian, H. L., "Smooth Surface Interpolation," J. Math. and Phys., Vol. 39, 1960.

35. Ahlberg, J. H., Nilson, E. N., Walsh, J. L., "The Theory of Splines and Their Applications," Academic Press, New York, 1967.

36. Johnson, R., Nowacki, H., Ogilvie, T. F., "Computer-Aided Ship Design," Lecture Notes, Engineering Summer Conference, The University of Michigan, Ann Arbor, May 1968.

37. Gass, S. I., "Linear Programming," McGraw-Hill Book Company, New York, second edition, 1964.

38. Jahnke, E., Emde, F., Lösch, F., "Tables of Higher Functions," McGraw-Hill, New York, 1960

39. Timoshenko, S., "Strength of Materials," Van Nostrand Company, Inc., Princeton, 1955.

40. Coons, S. A., "Surfaces for Computer-Aided Design of Space Figures," unpublished notes, Mechanical Engineering Department, M.I.T., January, 1964.

41. Kaiser, G., Nagel, J., Schiller, E., "Electronic Computation of System Measures," in German, Schiff und Hafen, 1968.

42. Forsythe, G. E., "Generation and Use of Orthogonal Polynomials for Data-Fitting with a Digital Computer," Journal of Industrial and Applied Mathematics, June 1957.

43. Cadwell, J. H., Williams, D. E., "Some Orthogonal Methods of Curve and Surface Fitting," The Computer Journal, 1962.

44. Ferguson, J., "Multivariable Curve Interpolation," Journal of the Association for Computing Machinery, Vol. 11, 1964.

2. Computer-Aided Preliminary Ship Design.

2.1 Definition of problem.

It is clear that we cannot begin to design something by computer before we know how to design it without computer. This does not mean that the computer is limited to conventional design procedures or may not be applied to innovative designs. On the contrary, the computer is encouraging the rational treatment of newly arising problems.

But we must keep in mind that writing a computer program requires an algorithmic description of the problem, this means a unique definition of the sequence of computations. This is not always easy in the case of design where many complex situations need to be evaluated and where human judgment is in its true element.

It will be the purpose of the following discussions to illustrate how the conventional design process, or at least most of its steps at this stage, can be simulated by computer, apparently without loss of validity. Subsequently, the issue of human versus computer efficiency in preliminary ship design will be given some thought.

There is no contention, of course, that the computer can generate ideas, a most essential ingredient of design. But the computer is indefatigable in evaluating and perfecting any number of combinations of given basic ideas. These ideas, i.e. basic configurations and operating concepts, the loosest possible description of the desired species ship, must be computer input. I leave it to the reader to ponder how many of the ideas we use in ship design are actually basic.

The purpose of preliminary ship design may be briefly stated as follows: "From a given set of functional requirements and in view of known environmental and legal constraints, determine the principal characteristics of the design (size, proportions, speed, etc.) so as to optimize a measure of merit of prescribed type."

A. The functional requirements vary considerably with the type of ship. For merchant ships the following set may be typical:

*Payload, if specified by the owner on the basis of a market analysis for trades of limited cargo availability (general cargo ships, container ships and the like). If unlimited amounts of cargo are available (tanker, dry bulk carrier) the payload would normally not be given as a requirement, but must be derived from the constraints (port depths, etc.),

*Speed, if specified by the owner on the basis of his operating schedule and market analysis. Otherwise, speed becomes a design variable, see below.

*Endurance, more precisely the greatest distance between refueling ports.

*Propulsion requirements, type of propulsion system, or alternatives under consideration (steam versus diesel, single versus twin screw, etc.).

*Mission requirements, such as cargo handling capabilities, extent of cargo refrigeration, maneuvering capabilities, and so on.

*Crew requirements, unless dictated by legal factors.

B. Constraints:

*Restrictions of waters and ports, limiting draft, beam, length and sometimes height of superstructures.

*Seakeeping constraints, imposing limits on ship motions and accelerations derived from considerations of ship strength, safety, and crew comfort.

*Safety constraints, pertinent to stability, freeboard, survival of damage, lifesaving equipment, fire hazards.

*Trim constraints, often introduced by the designer to ensure reasonable resistance properties, propeller submergence, and strength.

*Manning rules, where applicable by law.

C. Measures of merit:

Measures of merit relate benefit to effort. In merchant ship design, profit is the desired benefit, and the size of the investment measures the effort. Their ratio, the profitability of the investment, is to be maximized. There are many, more or less perfect, ways of expressing profitability. The following list suggests a few, for more details see Ref. 66.

*<u>Yield</u>, most direct measure of profitability, defined as after tax interest rate equivalent to the investment. May be determined by the Discounted Cash Flow method, Ref. 66, for predictable revenues and known costs.

*<u>Capital recovery factor</u>:

$$CRF' = \frac{A'}{P} \qquad (84)$$

where $A' = R - Y - T$ = return after taxes
R = revenue
Y = operating cost
T = taxes
P = investment

CRF' is of immediate usefulness only where revenues are predictable, returns are uniform, and lives of the designs are equal.

*<u>Net present value index</u>, an alternative to yield for known revenues:

$$NPVI = \frac{NPV}{P} \qquad (85)$$

NPV = net present value of all cash flows based on interest rate specified by investor, see Ref. 66.

*<u>Average annual cost</u>, useful where revenues are unknown, but the same for all designs:

$$AAC = Y + CR \cdot P \qquad (86)$$

Y = annual operating cost
CR = capital recovery factor for given lives and interest rates
$CR \cdot P$ = annual cost of capital recovery

*<u>Required freight rate</u>:

$$RFR = \frac{AAC}{C} \qquad (87)$$

C = annual transport capacity, a measure of benefit where revenues are unknown, but not the same among design alternatives

*Ship merit factor, as defined by Cheng, Ref. 67, is the inverse of RFR if one chooses to express the transport capacity in annual ton-miles. Among other formulations, Cheng gives the following breakdown:

$$SMF = k \frac{W_P \cdot V}{C} = \tag{88}$$

$$= 8760 \cdot f_s \cdot f_w \cdot f_v \frac{1}{1 + f_P} \frac{1}{C'} \frac{W_P}{W} \cdot \frac{1}{\frac{R}{W}} \cdot \frac{RV}{P_B}$$

where

SMF = ship merit factor

$k = 8760\ f_s \cdot f_w f_v/(1 + f_P)$ = service constant

W_P = payload

V = design speed in knots

C = AAC = average annual cost

$8760 = 24 \cdot 365$ hours/year

f_s = utilization factor, percent of annual service hours

f_w = load factor, percent of designed payload

f_v = operating speed factor, percent of design speed

f_P = port time factor, port time/sea time

$C' = C/P_B$ = specific operating cost in dollars per horsepower-year

P_B = power delivered by prime mover

W_P/W = payload-displacement ratio

R/W = drag-displacement ratio

$RV/P_B = \eta_O \cdot \eta_H \cdot \eta_R \cdot \eta_S$ = propulsive efficiency

$\eta_O, \eta_H, \eta_R, \eta_S$ = open water, hull, relative rotative and shaft transmission efficiencies, see Ref. 64.

The breakdown shows how technical and operational improvements affect the overall success of the design. See Ref. 67 for a more thorough discussion.

D. Design variables, also called decision variables, are typically

*Length, L

*Beam, B

*Draft, T

*Depth, D

*Block coefficient, C_B

*Midship coefficient, C_M, unless fixed

*Speed, V, unless specified by owner

Displacement, and prismatic coefficient are left out since they can be derived from the above. But they might replace two of the above since the role of dependent and independent variables is arbitrary.

An alternative set of design variables, dimensionless with one exception, is

$$L, \frac{L}{B}, \frac{B}{T}, \frac{L}{D}, C_B, C_M, \frac{V}{\sqrt{gL}}$$

In practice, there are sometimes fewer than seven unknowns if some of the variables are fixed by owner's requirement or due to constraints.

E. Intermediate results and output.

Before the measure of merit can be evaluated for any assumed set of principal dimensions the design must be worked out in some detail and its feasibility verified. This is discussed in the following section. The final output consists of the best set of design variables in the feasible range, and a crude description of the associated ship. A look at some of the second-best solutions is often also of advantage.

2.2 Elementary relations of preliminary design.

The purpose of this section is not to give a complete catalog of relations that have proven useful in preliminary design. The objective is simply to review some typical steps and relationships under the aspect of computer adaptability.

The flow chart in Fig. 15 gives an overview of the steps in preliminary design. There is, of course, a great deal of variation from design to design and from designer to designer as to the most suitable sequence of steps, usually leading to the same answer in a more or less straightforward manner. This is indicated in Fig. 15 by the staggered, incomplete loop arrows: After any stage it may be decided to go back a number of steps to correct infeasible results, or to skip any number of subsequent steps as a shortcut if it is intended to do a more complete analysis during the next checking

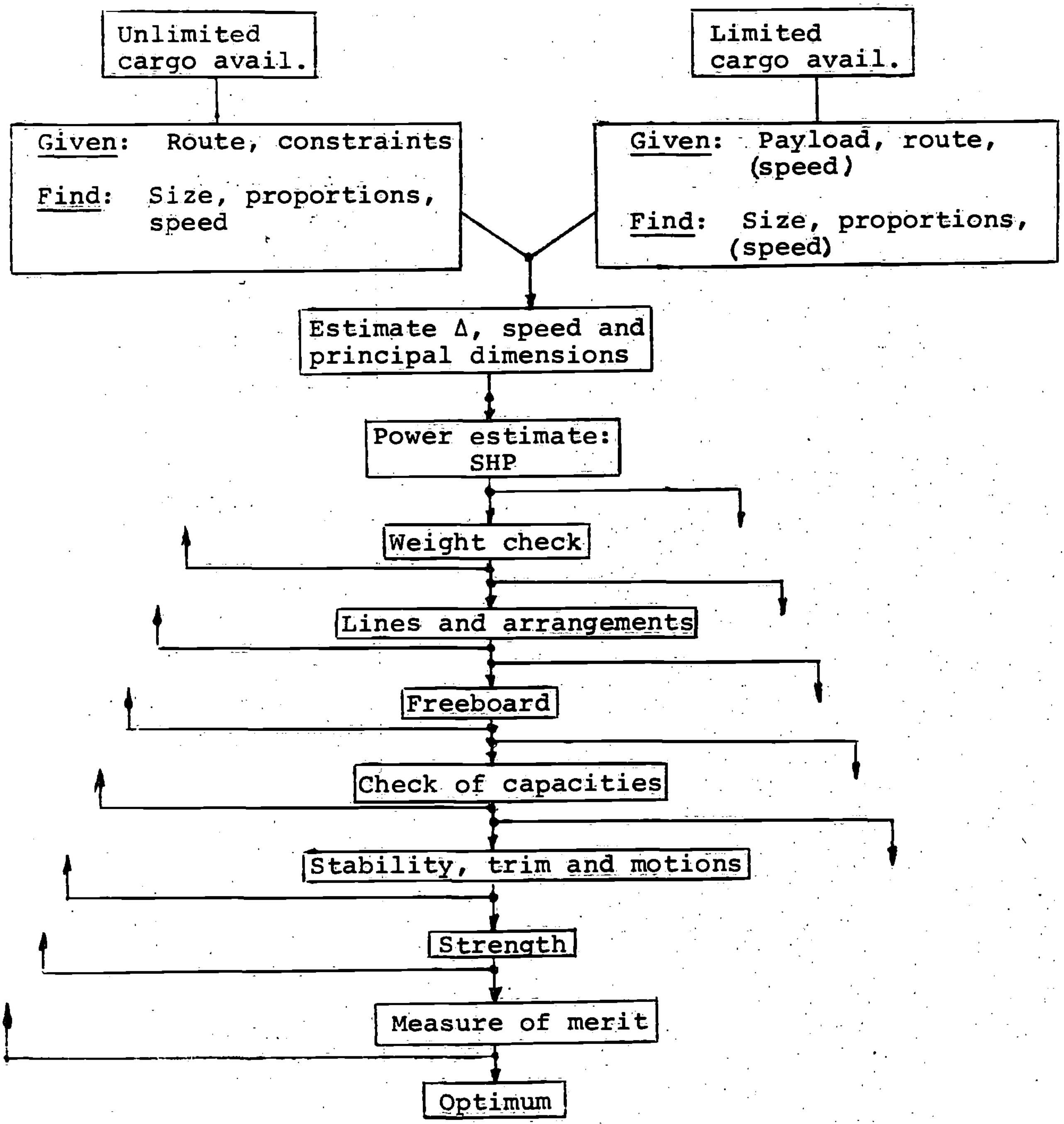

Unlimited cargo avail.
Limited cargo avail.
Given: Route, constraints
Find: Size, proportions, speed
Given: Payload, route, (speed)
Find: Size, proportions, (speed)
Estimate Δ, speed and principal dimensions
Power estimate: SHP
Weight check
Lines and arrangements
Freeboard
Check of capacities
Stability, trim and motions
Strength
Measure of merit
Optimum

Fig. 15: Flow chart of preliminary design

cycle. In the end, one will have a feasible design and its measure of merit. This does not preclude, however, that the measure of merit is used in earlier stages to reject some less desirable alternatives. An optimum can be found among the feasible designs as discussed in section 2.3. Let us now look at the individual steps of the design process.

2.21. Displacement estimate.

A good first guess of the displacement helps to find a feasible design fast, but is not absolutely critical since the displacement will be adjusted when weights are known more exactly.

In the case where payload is given the designer can fairly reliably estimate payload/deadweight, and deadweight/displacement ratios from similar ships. Hence

$$\Delta = \frac{\text{payload}}{\frac{\text{payload}}{\text{deadweight}} \cdot \frac{\text{deadweight}}{\text{displacement}}}$$

If cargo availability is unlimited, and physical constraints govern the size, an upper bound for the displacement can be constructed from assumptions as to

$$T_{max},\ \left(\frac{B}{T}\right)_{max},\ \frac{D}{T},\ \left(\frac{L}{D}\right)_{max},\ C_{B_{max}}$$

With some experience both methods furnish a suitable first guess which may serve as computer input.

2.22 First estimate of principal dimensions.

In the usual British units, the law of Archimedes requires

$$35\Delta = C_B\ LBT = C_P \cdot C_M\ LBT \qquad (89)$$

from which

$$B = \sqrt[3]{\frac{2240\Delta\ \frac{B}{T}}{C_B \cdot \frac{L}{B}}} \qquad (90)$$

and

$$T = \sqrt{\frac{2240\Delta}{C_B\ L \cdot \frac{B}{T}}} \qquad (91)$$

When Δ is known it is only necessary to estimate certain dimensionless form coefficients to find the principal dimensions by means of Eqs. 90 and 91.

For crude orientation, ranges covered by Taylor's Standard Series:

C_P	0.48 to 0.86
$\frac{B}{T}$	2.25 to 3.75
$C_V = \frac{35\Delta}{L^3}$	up to 0.007

Generally, the designer will be able to define narrower ranges within which the optimum should be sought, knowing by experience where infeasible solutions result. These ranges can be used by the computer in the optimum search, and the first guess selected accordingly. Further useful guidance:

*Curves of optimal $C_P = f\left(\frac{V}{\sqrt{L}}\right)$ after Taylor, Ref. 64, and after Van Lammeren.

*Curves of corresponding favorable values of C_B, C_P, C_M after Lap, Ref. 50.

*An equation for favorable C_M, given in Ref. 60 without further source:

$$C_M = 0.977 - 0.018\left(\frac{V}{\sqrt{L}}\right) + 0.76\left(\frac{V}{\sqrt{L}}\right)^2 - 0.115\left(\frac{V}{\sqrt{L}}\right)^3$$

*Favorable $C_B = 1.08 - 1.68\,F_n - 0.244\,F_n{}^2$, with $F_n = \frac{V}{\sqrt{gL}}$, after Van Lammeren.

2.23 Power estimate.

Since Δ, C_B, C_P, C_M, C_V, $\frac{L}{B}$, $\frac{B}{T}$ are now known from the previous step the effective horsepower can be estimated easily from

*Taylor Standard Series. Computer programs for this purpose are available. They are multi-variable interpolations of number tables based on the graphs of Ref. 48.

*Series 60, BSRA series, or others.

The bare hull predictions ought to be corrected for appendage drag.

The shaft horsepower:

$$\text{shp} = \frac{\text{ehp}}{\eta_D} = \frac{\text{ehp}}{\eta_O \cdot \eta_H \cdot \eta_R} \qquad (92)$$

The open-water efficiency, η_O, may be determined from:

*Troost series. Programs exist.

*From the equation

$$\eta_O = \eta_i \cdot \eta_\varepsilon$$

where $\eta_i = f(C_T, \lambda, Z, \varepsilon_m)$,

induced efficiency from Kramer diagram or corresponding equations

$$\eta_\varepsilon = f(\varepsilon_m, \lambda_i)$$

$$= \frac{1 - 2\varepsilon_m \lambda_i}{1 + \frac{2}{3}\frac{\varepsilon_m}{\lambda_i}} \text{, blade efficiency}$$

$C_T = \frac{T}{(0.5 \rho V_A{}^2 \cdot A_D)}$ = thrust loading coefficient.

$\lambda = \frac{V_A}{(\pi n D)}$ = advance ratio.

$\lambda_i = \frac{\lambda}{\eta_i}$ = induced advance ratio.

Z = number of blades.

$V_A = V(1 - w)$ = advance speed.

w = wake fraction.

A_D, D = propeller disk area, and diameter.

n = revolutions per second.

ε_m = mean drag-lift ratio of blade elements, estimated according to imminence of cavitation, between 0.04 and 0.08, safer at upper limit.

For both estimating methods, wake and thrust deduction fraction must be given, as will be explained. The number of revolutions may be optimized within specified limits.

The hull efficiency, η_H:

$$\eta_H = \frac{1 - t}{1 - w}$$

where t = thrust deduction fraction.

w = wake fraction.

Estimate for t, and w may be obtained from

*Series 60.

*Approximate equations.

For example, from diagrams of Harvald, Ref. 46, I obtained by least square approximation:

For single-screw ships:

$$w = k_1(-0.10 + 0.56\ C_B + 2.6(C_B - 0.6)^2) + k_2 + k_3 \qquad (93)$$

where $k_1 = \dfrac{7}{\frac{L}{B}\left(1 + (0.13 - 0.14\ C_B)\left(7 - \frac{L}{B}\right)\right)} = \frac{L}{B}$ - correction.

$k_2 = f_s(-0.0625 + 0.15\ C_B)$ = section shape correction.

f_s = -1 to +1 from extreme V to extreme U sections.

$k_3 = -4.85\left(\frac{D}{L} - 0.04\right) + 90\left(\frac{D}{L} - 0.04\right)^2$ = diameter correction.

D = propeller diameter.

For twin-screw ships:

$$w = k_4(-0.1 + 0.35\ C_B + 16.4(C_B - 0.6)^2) \qquad (94)$$

where $k_4 = \dfrac{7}{\frac{L}{B}\left(1 + (0.72 - 1.2\ C_B)\left(7 - \frac{L}{B}\right)\right)} = \frac{L}{B}$ - correction

After Schoenherr, Ref. 64,

for single-screw ships

$t = (0.5 \text{ to } 0.7)w$, depending on rudder resistance,

for twin-screw ships

with bossings: $t = 0.14 + 0.25\ C_B$

with struts: $t = 0.06 + 0.7\ C_B$

The relative rotative efficiency:

$\eta_R \approx 1$, for purposes of preliminary design.

Service allowance:

Finally, according to U.S. design practice a 25 percent power margin is usually added to the calm water trial shaft horsepower to allow for speed losses due to the seaway and fouling.

2.24 Weight check.

With ship displacement, hull proportions, and machinery power established, a refined weight estimate can be obtained, traditionally based on the following weight categories:

Light ship:

Hull structure
Outfitting and hull engineering
Machinery

Deadweight:

Payload
Fuel
Fresh water (feed and domestic)
Lubricating oil
Provisions and stores
Passenger, crew, and effects
Dunnage

Estimating relationships for these categories may be based on

*Empirical relationships published in the literature, for example Refs. 51 to 59, and 65. In these relationships weight equations are given in terms of the significant hull form and operational parameters.

*Shipyard weight data. The relationships derived from such data are in principle the same as above.

*Classification rules. The use of ABS or other rules requires a fair amount of detailed knowledge about the design.

Computer programs for steel weights based on the rules have been written by Evans and Khoushy, Ref. 54. While such programs may bring about desirable refinements at a somewhat later stage of design, they are too time-consuming for early weight estimates.

As an illustration for parametric weight equations, let us quote a few examples from Refs. 59 and 60 for a general cargo ship. Equations of this type are, of course, easily programmed.

Steel weight:

$$W_s = 2.107 \left[\frac{L(B + D) \cdot 0.986}{100} \right]^{1.19} \qquad (95)$$

Outfit and hull engineering:

$$W_O = 0.15 \left[\frac{(L \cdot B) \cdot 0.986}{100} \right]^{1.19} \qquad (96)$$

Machinery, wet:

$$W_M = 7.18 \ SHP^{0.495} \qquad (97)$$

Fuel:

$$W_F = \frac{1.1 \ E \cdot SHP \cdot FR}{2240 \cdot V}, \text{ long tons} \qquad (98)$$

where $FR = 0.5 \frac{SHP}{SHP - 855}$ = fuel rate, steam propulsion, in lbs per HP-hr.

1.1E = refueling range in miles, with 10 percent allowance.

V = speed in knots.

If the weight total does not add up to the displacement originally estimated one will normally go back and try another displacement estimate, suitably the one just obtained as weight total.

2.25. Lines and arrangements.

Preliminary lines and arrangement sketches may serve two purposes in the early stages of ship design, namely to check:

*The feasibility of the selected set of principal dimensions as to freeboard, capacities, stability, trim, etc.

*The "optimality" of these dimensions, this means the effect of lines and arrangements on the measure of merit.

On the one hand, even crude lines and arrangement plans are somewhat laborious to prepare, and it is tempting to by-pass this step until after the principal dimensions are well established. This can sometimes be justified where the feasibility of the design is beyond doubt on the basis of past experience and where the effect of the arrangements on cost and effectiveness of the ship can be neglected. But, on the other hand, one need only look at a container ship or military vessels to realize that frequently lines and arrangements have a decisive influence on the functional capabilities and hence on the measure of merit.

Nevertheless, it is probably fair to say that lines and arrangement design is one of the weakest spots of computer-aided preliminary design. Even when using a computer one tries to avoid the disproportionate effort involved in this step. But several other possibilities exist as listed in the following overview.

Lines design:

*By-pass: As discussed, this is permissible where the design study may be based strictly on principal dimensions.

*From systematic series (Taylor, 60, BSRA, Lap): Where resistance data are taken from systematic series it is logical to use the associated lines as dummies in the design study.

*From similar ships: If the scope of variation in hull form is moderate hull shapes may be derived by systematic distortion of a parent shape supplied as input. A computer program for this purpose was prepared by the Maritime Administration using the methods of Lackenby, Ref. 6. Gallin, Ref. 62, uses a similar approach.

*From manual design: The input may be a manually designed parent. This is particularly suitable for simple shapes, for example a prismatic, hard-chine planing hull.

*From hull parameters: This approach corresponds to the elaborate methods for lines creation reviewed in section 1. Hamilton and Weiss, Ref. 24, have simulated manual design working from a sectional area curve with the aid of Coons' surfaces.

Arrangement design:

*By-pass: If the arrangement specification step is by-passed only relatively crude feasibility checks will be possible by means of parametric equations based on principal dimensions.

*Input: Most frequently, one type of arrangement is assumed for the study and read in as input. For example, the number of decks, bulkheads, superstructures and their relative location and extent are fixed.

*Input of general topology: To provide more choice the input may specify only a loose conceptual definition without fixing the number of decks, bulkheads, and similar details. The program can then perform a sub-optimization of the topology for every set of principal dimensions. This method is followed in the destroyer feasibility studies of Ref. 61.

2.26. Freeboard.

The available freeboard is measured from the freeboard deck

$$F_A = D_F - T \qquad (99)$$

D_F = depth of freeboard deck at deck line

T = draft

The required freeboard, according to the 1966 International Load Line Convention, is for the summer load line:

$$F_R = (F_{TAB} + k_L) \cdot k_{C_B} + k_D + k_S + k_{Sh} \quad (100)$$

where

F_{TAB} = tabular freeboard, depending on ship type

k_L = length correction, necessary for ships of up to 328 ft. length if the superstructures are shorter than 35 percent of the length

$k_{C_B} = \frac{C_B + 0.68}{1.36}$ = correction for C_B

k_D = depth correction, necessary if $L/D < 15$.

k_S = superstructure correction

k_{Sh} = sheer correction

The tabular freeboard can be easily approximated by a polynomial, see Ref. 60. All other terms involved are even simpler. In refining the design, we must also account for:

*Freeboard reductions obtainable when meeting one-compartment flooding standard

*Minimum bow height

*Miscellaneous other details of the rules.

2.27. Capacities.

For first estimates, one finds the gross bale cubic

$$GBC = C_{BC}(L \cdot B \cdot D \cdot C_B) \quad (101)$$

where

C_{BC} = coefficient for bale cubic, found in Ref. 64, page 284.

The payload volume and stowage factor are

$$V_{PAYL} = GBC - V_{MS} - \left\{\frac{W_F}{\gamma_F} - (V_{FB} + V_{FS})\right\} \quad (102)$$

$$SF = \frac{V_{PAYL}}{W_{PAYL}} \quad (103)$$

where

V_{MS}, V_{FB}, V_{FS} = volume of machinery space, double bottom fuel capacity, and settler tank fuel capacity

W_F = fuel weight from section 2.24

γ_F = specific weight of fuel, long ton/cu. ft.

In later design stages more accurate capacity checks are obtained by integration according to lines and arrangements, see Refs. 62 and 68.

2.28. Stability, trim and motions.

The requirements of transverse stability serve primarily to ensure a minimum safety against capsizing (minimum metacentric height, and range of stability), and secondarily, to avoid excessive roll accelerations (maximum metacentric height).

The minimum metacentric height, GM, is therefore governed the requirements to withstand heeling moments due to beam winds, asymmetric weight distributions (such as by passengers crowding to one side), lateral tow rope pull, centrifugal forces in a turning circle, handling of heavy cargo far outboard, etc. Free liquids on board must be accounted for. Simple estimating relationships for these effects are found in the design literature, for example Ref. 64.

A check of the range of stability requires more elaborate computations, but can usually be deferred to the later stages of design unless the freeboard is critically low.

Where the rules for damage stability must be met (2 in. residual GM, 3 in. residual freeboard, angle of heel in asymmetric flooding limited) it is also convenient practice to allow margins in early design derived from similar ships, and to check more exactly later.

An upper bound for the initial stability so as to avoid excessive roll accelerations, or short roll periods can be specified in terms of a minimum (reasonable) natural period of roll, τ_{min}, or

$$GM \leq \frac{4\pi^2 \cdot (kB)^2}{g\, \tau_{min}^2} \qquad (104)$$

where

kB = roll gyradius, average about 0.4 B, see Ref. 64.

The stability requirements, upper and lower bounds, are compared to the actual metacentric height of the design. This needs to be done for several loading conditions of the ship as may be critical for the bound in question:

Light ship
Full load, homogeneous cargo
Full load, cargo on deck
Full load, heaviest cargo in hold
Ballast conditions

Depending on the case, departure and/or end of voyage conditions need to be examined.

The actual GM = KB + BM - KG is generally estimated on the basis of parametric equations for its components. For KG, the same weight categories as for the weight estimates, section 2.24, ought to be used, for example for the steel: $KG_S = k_S \cdot D$, k_S = a constant, D = depth. The individual vertical centers are then combined into the total KG. For some estimating relationships, see Ref. 59.

The quantities KB and BM depend on the hull geometry, and any parametric equations must be limited to their pertinent ship type. Examples are found in Refs. 59, 60, 64.

Trim is of influence on resistance and propulsion (propeller submergence), seakeeping, and maneuverability. Only some of these effects can be assessed quantitatively in early design so that the designer must be satisfied with specifying trim as a constraint, except that resistance penalties may be introduced where they are known from similar ships. Trimming moments may be estimated, given the capacity curves and LCB. The trim is then approximated using the moment to trim one inch.

The influence of ship motions upon the design goes, of course, far beyond the simple roll period effect mentioned above. As our fundamental knowledge of speed losses and motions in a seaway increases it becomes possible to allow for such effects in design studies assisted by the computer, whereas in the past the work involved in systematic evaluations would have been prohibitive. Steps in this direction, the direction of a more rational and comprehensive treatment of ship motions in design, are taken in Refs. 69, and 74. (Analogously, regarding ship vibrations, Ref. 70 presents an approach to rational design decisions based on computer studies.)

2.29. Strength.

Where classification rules apply and conventional structural design is used strength requirements play no direct part in preliminary design although their effects are present in the weight estimates. Even where strength is a decisive factor for the feasibility of the design a constraint on L/D as input will be sufficient at first, and the structural suboptimization may be carried out later.

2.210. Cost.

Cost estimates are needed for the measure of merit evaluation, section C. Generally, both investment and operating cost are required. Investment cost estimates are usually related to the weight categories listed in section 2.24, i.e., steel, outfit, and machinery cost are estimated separately.

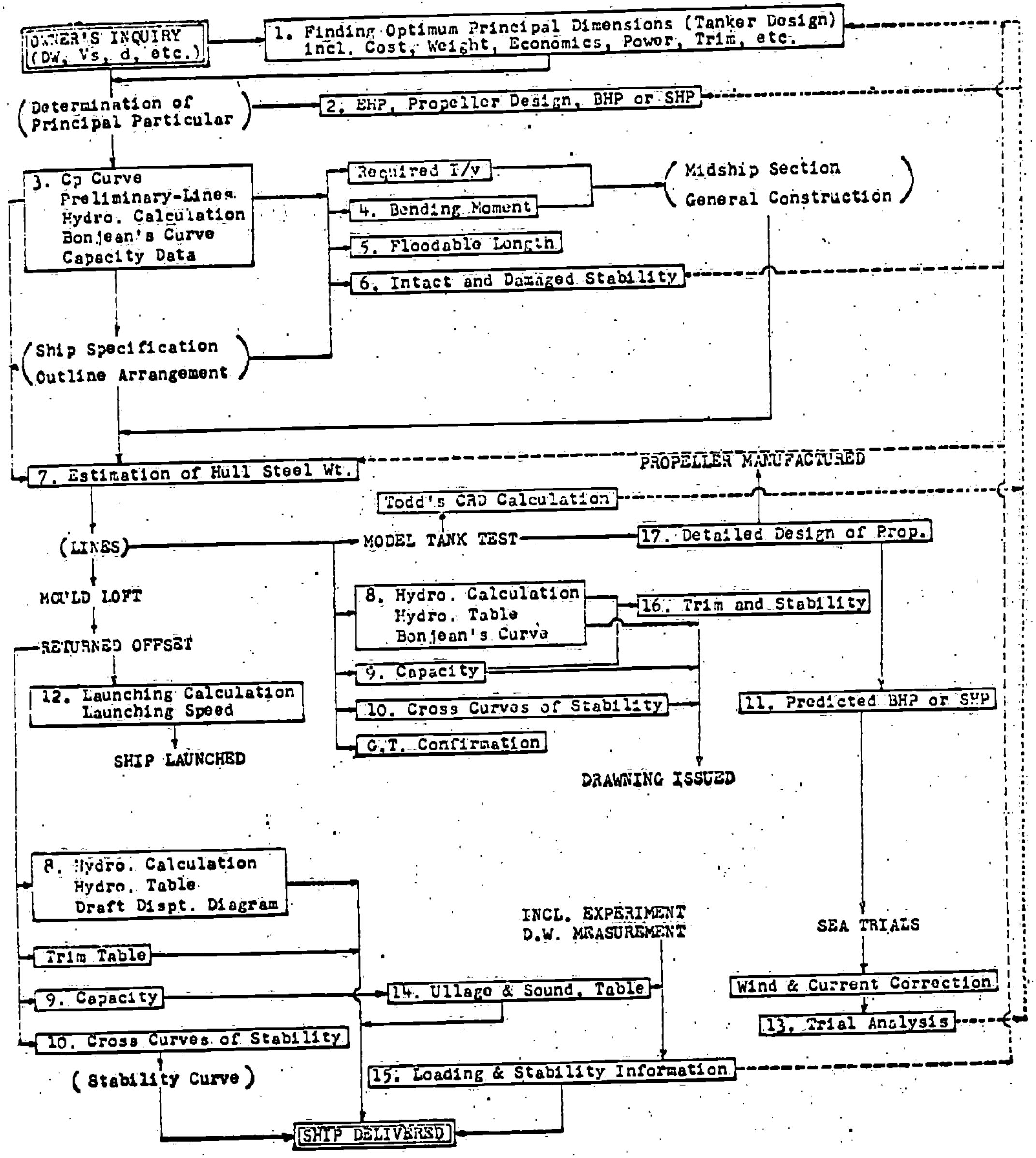
Current U.S. shipbuilding cost data are well presented in Ref. 71. For many other important parametric cost data, see Refs. 58, and 65, and the references quoted therein.

As a special annotation, it may be mentioned that fuel cost must be adjusted to ship use, i.e., operating times in full load, ballast, and port. For some details of this procedure, see Ref. 60.

2.3. Examples.

2.31. Yamagata and Akatsu (1964).

The authors, associated with Mitsubishi in Yokohama, were among the first to publish the description of a system of preliminary and detail design programs, Ref. 57. The flow chart, Fig. 16, gives an overview. The preliminary design part is only sparsely described. Here are some highlights.


OWNER'S INQUIRY (DW, Vs, d, etc.)
1. Finding Optimum Principal Dimensions (Tanker Design) incl. Cost, Weight, Economics, Power, Trim, etc.
(Determination of Principal Particular)
2. EHP, Propeller Design, BHP or SHP
3. Cp Curve Preliminary-Lines Hydro. Calculation Bonjean's Curve Capacity Data
Required I/y
4. Bending Moment
5. Floodable Length
6. Intact and Damaged Stability
(Midship Section General Construction)
(Ship Specification Outline Arrangement)
7. Estimation of Hull Steel Wt.
PROPELLER MANUFACTURED
Todd's CAD Calculation
(LINES)
MODEL TANK TEST
17. Detailed Design of Prop.
MOULD LOFT
RETURNED OFFSET
8. Hydro. Calculation Hydro. Table Bonjean's Curve
16. Trim and Stability
9. Capacity
10. Cross Curves of Stability
G.T. Confirmation
11. Predicted BHP or SHP
12. Launching Calculation Launching Speed
SHIP LAUNCHED
DRAWNING ISSUED
8. Hydro. Calculation Hydro. Table Draft Dispt. Diagram
Trim Table
9. Capacity
10. Cross Curves of Stability
(Stability Curve)
14. Ullage & Sound. Table
INCL. EXPERIMENT D.W. MEASUREMENT
SEA TRIALS
Wind & Current Correction
13. Trial Analysis
15. Loading & Stability Information
SHIP DELIVERED

Note:

1. Items in parenthesis show the works by manual calculation.
2. Items in single frame show computer system developed, and number therein shows major program as listed on Table 1.
3. Full line and arrow shows routine work flow.
4. Dotted line and arrow shows analyzing and data feed-back procedure.

Fig. 16: Flow chart of design process, from Yamagata and Akatsu, Ref. 57.

Application: Tanker design.

Given: Payload, speed, endurance, draft limit.

Design variables: L, B, T, D, C_B.

Measure of merit: Cost (?).

Subprograms: Load lines
Steel weight by midship section modulus from rules
Economics
Preliminary lines from standard series.

2.32. Hamilton and Weiss (1965).

Reference: 24.

Affiliation: MIT.

Objective: Primarily lines design, although aiming at the whole process shown in Fig. 17.

Application: Destroyer.

Given: L, B/T, C_P, C_M, C_V.

Special features: Sectional area curve, and lines by Coons' method. Unique procedure, no built-in optimization.

2.33. Murphy, Sabat, and Taylor (1965).

Reference: 59.

Affiliation: Maritime Administration.

Objective: Optimization of principal dimensions.

Application: General cargo ship.

Given: Payload, speed, endurance, stowage factor, general constraints.

Design Variables: $V/\sqrt{L}$ (hence L), B/T, C_P, Δ, L/D.

Measure of merit: Average annual cost.

Optimization method: The optimization method is an interesting adaptation of manual design techniques, see Fig. 18. Four variations of every design variable are first computed (1024 designs). The given constraints are then exploited, and suboptimizations against one design variable at a time are performed to "interpolate" for the least cost ship. This way the specified range

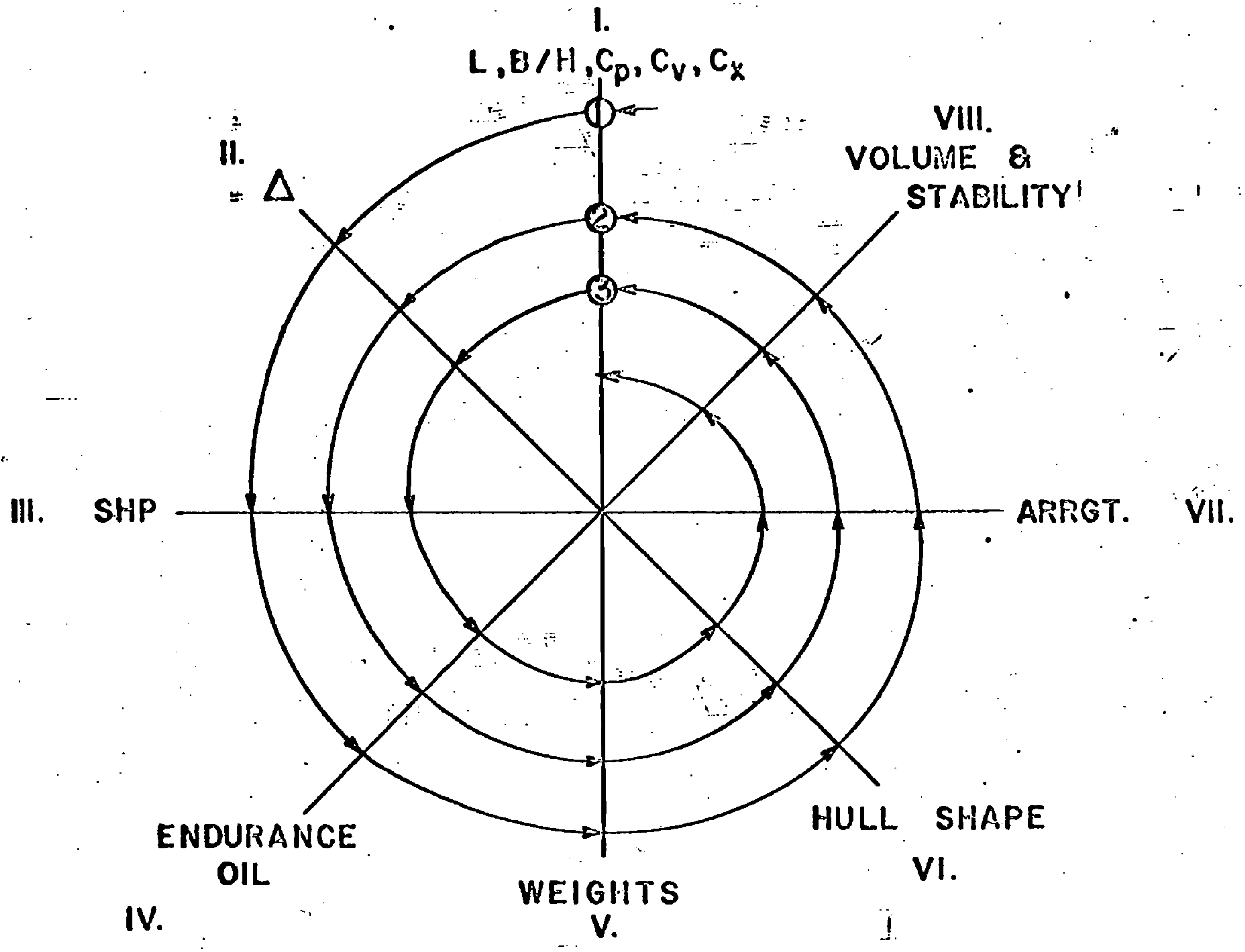


Fig. 17: Design spiral, from Hamilton and Weiss, Ref. 24.

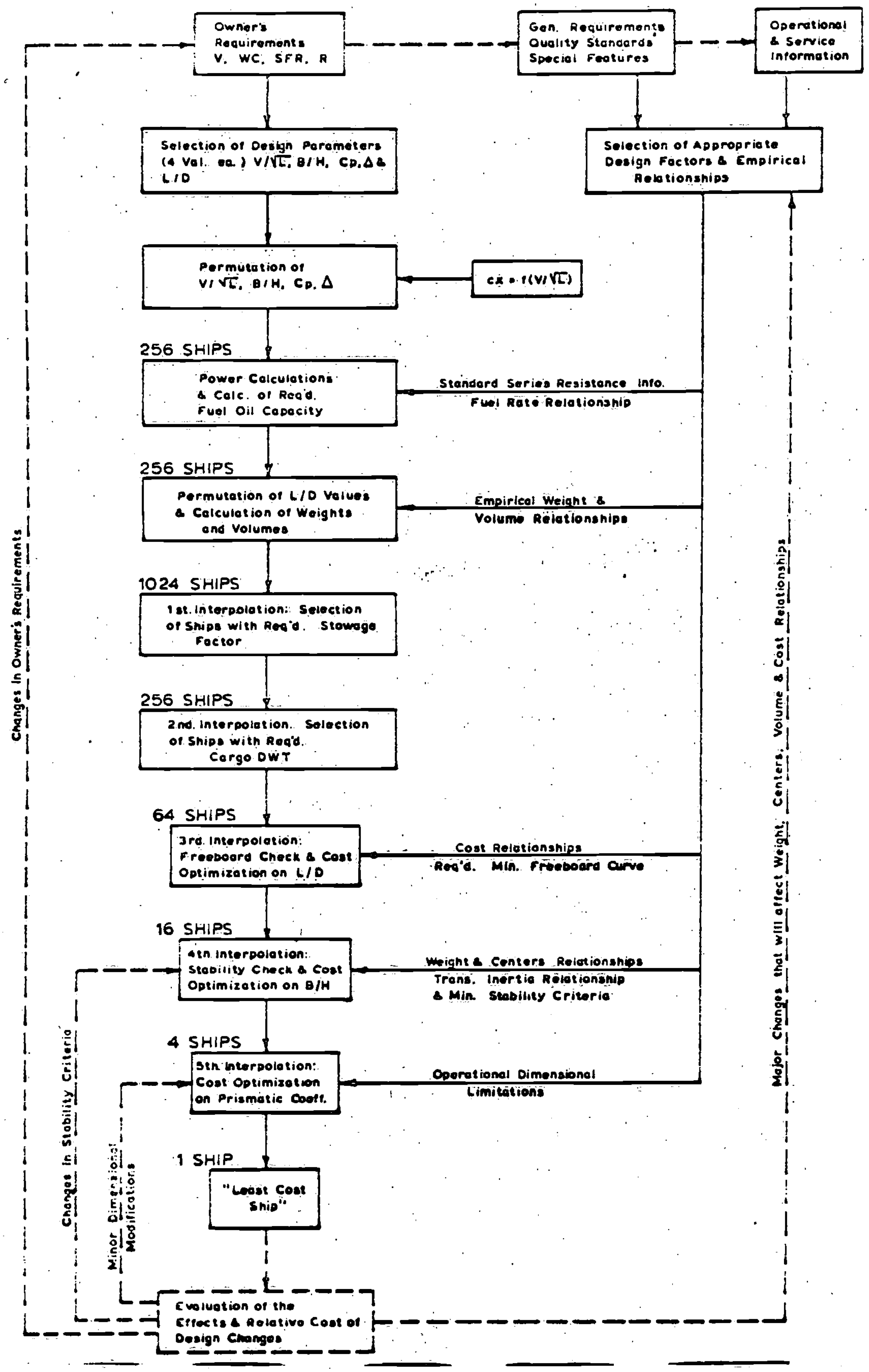

Owner's Requirements V, WC, SFR, R
Gen. Requirements Quality Standards Special Features
Operational & Service Information
Selection of Design Parameters (4 Val. ea.) V/√L, B/H, Cp,Δ & L/D
Selection of Appropriate Design Factors & Empirical Relationships
Permutation of V/√L, B/H, Cp, Δ
cx = f(V/√L)
256 SHIPS
Power Calculations & Calc. of Req'd. Fuel Oil Capacity
Standard Series Resistance Info.
Fuel Rate Relationship
256 SHIPS
Permutation of L/D Values & Calculation of Weights and Volumes
Empirical Weight & Volume Relationships
1024 SHIPS
1st. Interpolation: Selection of Ships with Req'd. Stowage Factor
256 SHIPS
2nd. Interpolation. Selection of Ships with Req'd. Cargo DWT
64 SHIPS
3rd. Interpolation: Freeboard Check & Cost Optimization on L/D
Cost Relationships
Req'd. Min. Freeboard Curve
16 SHIPS
4th Interpolation: Stability Check & Cost Optimization on B/H
Weight & Centers Relationships
Trans. Inertia Relationship
& Min. Stability Criteria
4 SHIPS
5th Interpolation: Cost Optimization on Prismatic Coeff.
Operational Dimensional Limitations
1 SHIP
"Least Cost Ship"
Evaluation of the Effects & Relative Cost of Design Changes
Changes in Owner's Requirements
Changes in Stability Criteria
Minor Dimensional Modifications
Major Changes that will affect Weight, Centers, Volume & Cost Relationships

Fig. 18: Least cost ship study-procedure schematic from Murphy, Sabat, and Taylor, Ref. 59.

of every design variable is searched exhaustively in about 4 min. of IBM 7090 computer time. The parametric equations used in this design model are obviously of the somewhat simpler variety. But in view of the well-known flatlaxity of the ship design optimum it may safely be assumed that a feasible design anywhere near the least cost ship will exhibit good economic success.

2.34. Stephans, Kreitner (1966,...).

References: 61, 63, and 36.

Affiliation: U.S. Navy.

Objective: Feasibility design for purposes of the U.S. Navy.

Applications: Destroyers, submarines.

Given: Payload weight and volume, speed, endurance, type of propulsion plant, constraints. Acceptable arrangements. For details, see Ref. 63.

Design variables: Principal characteristics, arrangements.

Measure of merit: Cost and effectiveness.

Optimization method: Exponential random search, see Refs. 60, and 36 (Stephans). See also section 3.

2.35. Mandel, and Leopold (1966).

Reference: 60.

Affiliation: MIT.

Objective: Optimization in preliminary ship design.

Applications: General cargo ship, tanker.

Given: Payload, speed, endurance, stowage factor, constraints.

Design variables: $V/\sqrt{L}$ (hence L), B/T, C_P, Δ, L/D.

Measure of merit:

Cargo ship:

$$C = w_1 (AAC)^2 + w_2 (W_{P_{REQU}} - W_{P_{ACT}})^2 + + w_3 (V_{P_{REQU}} - V_{P_{ACT}})^2 \qquad (105)$$

where $W_{P_{REQU}}$, $W_{P_{ACT}}$, $V_{P_{REQU}}$, $V_{P_{ACT}}$ = required and actual payload weight and volume.

w_1, w_2, w_3 = weighting factors, selected by designer according to priorities of terms.

Tanker:

$$C = CRF \left\{ 1 - 10 \left(\frac{\left| W_{P_{REQU}} - W_{P_{ACT}} \right|}{W_{P_{REQU}}} \right) \right\} \qquad (106)$$

Optimization method: Exponential random search. See section 3.

2.36. St. Denis (1966).

Reference: 72.

Affiliation: Naval Engineering Science Company.

Objective: Selection of size and principal characteristics.

Application: Aircraft carrier.

Given: Missions, functional requirements, constraints.

Design variables: Principal characteristics, arrangements.

Measure of merit: Cost and effectiveness.

Optimization method: Not specified in detail. But it is of particular interest to study the stochastic aspects of the design model.

2.37. Gallin (1967).

Reference: 62.

Affiliation: Blohm and Voss, Hamburg.

Objective: Systematic preliminary ship design and economic sensitivity analysis.

Application: Cargo ship.

Given: Deadweight, design or operating speed, endurance, stowage factor, constraints on T, GM, freeboard (rules), trim, L/B, B/T.

Design variables: Principal characteristics.

Measure of merit: Cost.

Optimization method: No search, just a comprehensive sensitivity analysis.

2.38. Kuniyasu (1968).

Reference: 68.

Affiliation: Ishikawajima-Harima Heavy Industries.

Objective: Optimization of principal dimensions.

Application: Tankers, bulk carriers, and combination carriers.

Given: Payload or deadweight, speed, endurance, stowage factor, constraints on T, B, SHP, and combinations thereof.

Design variables: Principal characteristics.

Measure of merit: Capital recovery factor, or annual operating cost per ton.

Optimization method: Output contour plot. Computational steps, see Fig. 19.

2.39. Boumis (1968).

Reference: 73.

Affiliation: Litton Systems, Inc.

Objective: This article does not describe any specific optimization method applied to ship design. It rather suggests the use of a macrocompiler, NEAT (Naval Engineering Analysis Technique), to assist in the formulation of ship design optimization problems. The NEAT processor accepts a NEAT program, written in a problem-oriented language under development at Litton, and translates the logic of these instructions into a regular optimization program (in FORTRAN). The NEAT formulation of the design problem then simply consists of the definition of the special design problem on hand, and the NEAT processor, from a general knowledge of design logic, works out the flow chart of the optimization problem to be created. It also makes reference automatically to the proper files in a data bank.

An approach such as this seems promising in order to resolve the conflict between the ever varying formats of ship design and the need to create computer programs in a cost-effective manner.

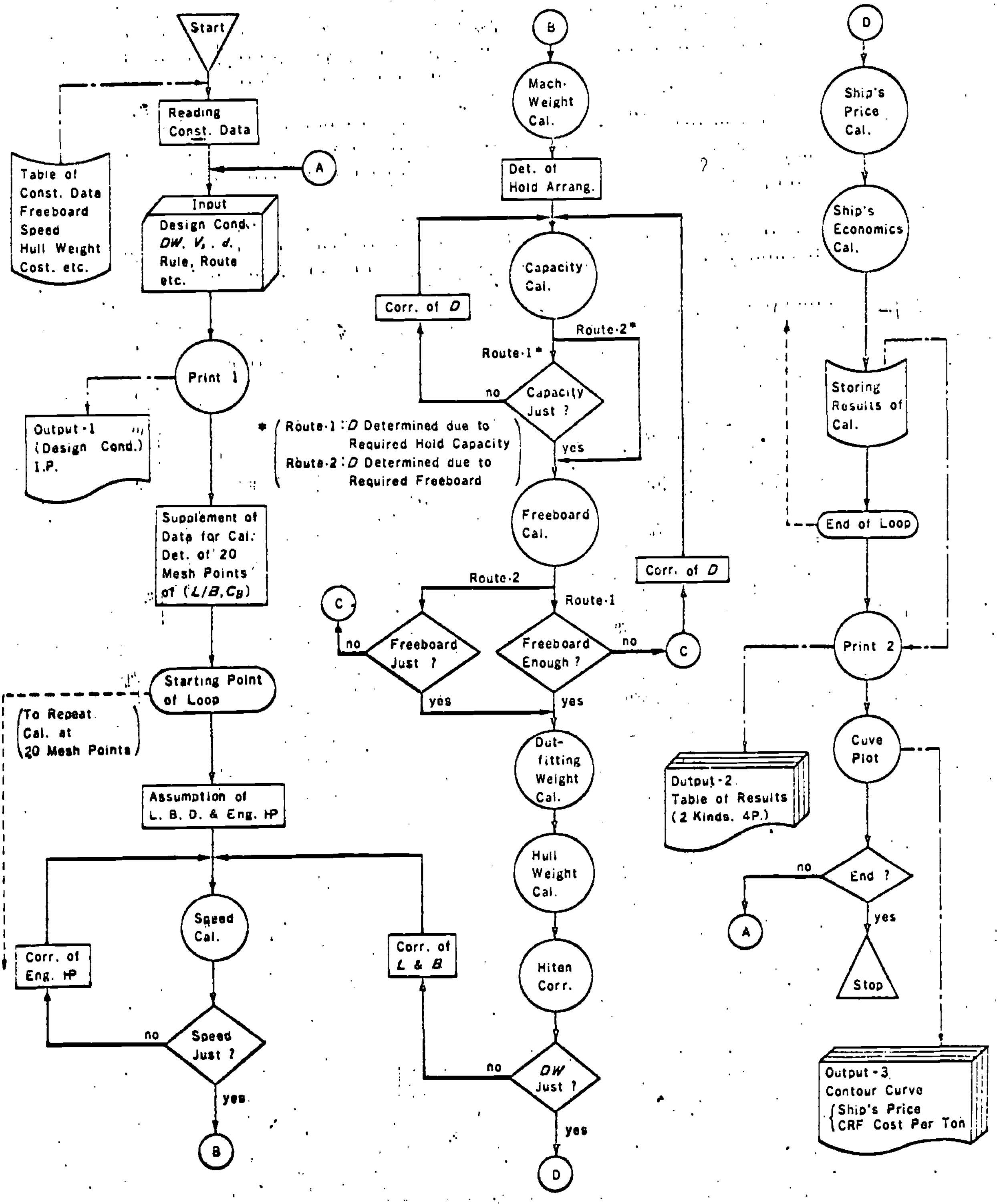


Fig. 19: Flow chart of preliminary design from Kuniyasu, Ref. 68.

2.4. Preliminary design as optimization problem.

To the extent that design permits choice it calls for optimization. It is true that in many cases the shipping line and the designing naval architect may share the responsibility for the preliminary design process defined in Fig. 15. The shipping line may generate from its functional needs certain technical requirements (say, deadweight and speed), and finally judge by its own standards the alternatives offered by the designer. In this situation, the designer acts only as a technical agent. But he will still aim to perfect his design within the scope of choice left to him. How technical success relates to economic success is illustrated in Cheng's Ship Merit Factor, Eq. (88).

There are also clear indications that the computer is broadening the scope of preliminary design gradually so as to require a demonstration of a degree of optimality where feasibility was sufficient in the past. This is certainly the trend in naval ship procurement policies, and can also be observed in the efforts of certain shipyards to develop a marketing organization for their standardized products.

It should be evident from the foregoing review of ship design relationships that this complex engineering task cannot easily be transformed into a standard mathematical form. But upon closer examination it becomes apparent that the various formats of the design problem can all be reduced mathematically to the general optimization problem:

Optimize the measure of merit function

$$F = F(x_1, \ldots\ldots, x_n)$$

$x_1, \ldots\ldots, x_n$ = design variables, subject to the inequality constraints

$$g_m = g_m(x_1, \ldots, x_n) \geqslant 0,$$

and the equality constraints

$$h_j = h_j(x_1, \ldots, x_n) = 0, j < n.$$

The functions F, g_m, and h_j are in general nonlinear in $x_1, \ldots, x_n$, and we have therefore obtained a general nonlinear programming formulation for the design problem. The implications of this result are discussed further in the following sections where some solution techniques of general nature are also presented.

2.5. References 45 through 74.

45. Stevens, E. A., "Machinery Weights," Marine Engineering and Shipping Review, 1942/43.

46. Harvald, S., "Gegevens ter bepaling van het volgstroom- and zoggetal van zeegaande enkelschroefschepen," Schip en Werf, 1949.

47. Ridgely-Nevitt, C., "Development of Graphic Aids to Preliminary Design," Journal ASNE, 1950.

48. Gertler, M., "A Re-Analysis of the Original Test Data for the Taylor Standard Series," TMB Report 806, Washington, 1954.

49. Munro-Smith, R., "Ship Design—Preliminary Determination of the Dimensions and Other Technical Characteristics," The Shipbuilder and Marine Engine-Builder, 1956.

50. Lap, A. J. W., "Fundamentals of Resistance and Propulsion," Part A, publication no. 129a of the N.S.M.B., Wageningen, Holland, 1956.

51. Powell, S. C., "Estimation of Machinery Weights," SNAME, 1958.

52. Dankwardt, E., "Determination of Displacement of Dry-Cargo Ships in the Project Stage," in German, Schiffbautechnik, 1961.

53. Watson, D. G. M., "Estimating Preliminary Dimensions in Ship Design," Institution of Engineers and Shipbuilders in Scotland, 1962.

54. Evans, J. H., Khoushy, D., "Optimized Design of Midship Section Structure," SNAME, 1963.

55. Illies, K., "Contribution to the Issue of Steam Versus Diesel Propulsion for Merchant Ships," in German, Transactions STG, 1963.

56. Munro-Smith, R., "Merchant Ship Design," Hutchinson, London, 1964.

57. Yamagata, A., Akatsu, N., "On the Application of Digital Computers to Ship Calculation and Initial Design Problems," Zosen Kiokai, 1964.

58. Johnson, R. P., Rumble, H. P., "Weight, Cost and Design Characteristics of Tankers and Dry-Cargo Ships," Marine Technology, April 1965, also Int. Shipbu. Progress, August 1964.

59. Murphy, R. D., Sabat, D. J., Taylor, R. J., "Least Cost Ship Characteristics by Computer Techniques," Marine Technology, April 1965.

60. Mandel, P., Leopold, R., "Optimization Methods Applied to Ship Design, SNAME, 1966.

61. Stephans, O. L., "New Developments in Feasibility Design Programs," Computer-Aided Ship Design and Construction Symposium, August 1968, Washington, D.C.

62. Gallin, C., "Design of Economic Ships by Means of Digital Computer," Transactions Schiffbautechnische Gesellschaft, 1967.

63. Kreitner, C. W., "Destroyer Feasibility Design Program," Computer-Aided Ship Design and Construction Symposium, August 1966, Washington, D.C.

64. Comstock, J. P., editor, "Principles of Naval Architecture," SNAME, New York, 1967.

65. Benford, H., "The Practical Application of Economics to Merchant Ship Design," Marine Technology, January 1967.

66. Benford, H., "Fundamentals of Ship Design Economics," Lecture Notes, The University of Michigan, 1968.

67. Cheng, H. M., "Performance Comparisons of Marine Vehicles," SNAME, New York Metropolitan Section paper, September 1968.

68. Kuniyasu, T., "Application of Computer to Optimization of Principal Dimensions of Ships by Parametric Study," Japan Shipbuilding and Marine Engineering, 1968.

69. Abkowitz, M. A., Vassilopoulos, L., Sellars, F. H., "Recent Developments in Seakeeping Research," SNAME, 1966.

70. Nowacki, H., "On the Steady-State Ship Hull Response," Tecnologia Naval, Rio de Janeiro, 1969, in press.

71. Fetchko, J. A., "Methods of Estimating Investment Costs of Ships," Lecture Notes, Engineering Summer Conferences, The University of Michigan, Ann Arbor, June 1968.

72. St. Denis, M., "The Strike Aircraft Carrier: Considerations in the Selection of Her Size and Principal Design Characteristics," SNAME, 1966.

73. Boumis, T. P., "Computer Definition of Ship Characteristics," Marine Technology, July 1968.

74. Hamlin, N., Compton, R., "Assessment of Seakeepability," Marine Technology, October 1966.

3.3 Notes on selected optimization techniques.

3.31 The method of Lagrangian multipliers.

One of the classical and most elegant optimization methods was devised by Lagrange. It is often called the method of undetermined (or Lagrangian) multipliers, and it serves to solve nonlinear optimization problems with equality constraints. (See Ref. 75 for a thorough exposition).

Given the object function

$$F = F(x_1, \dots, x_n) \tag{107}$$

whose extremum (maximum, minimum) we want to find, subject to the equality constraints (also called auxiliary conditions)

$$h_j = h_j(x_1, \dots, x_n) = 0 \, , \; j < n . \tag{109}$$

Both, the object function and the constraints may be nonlinear in the design variables $x_1, \dots, x_n$.

Obviously, we could use the equations (109) to eliminate j unknowns in (107). The remaining (n-j) unknowns could be found by solving for a stationary value of F from the conditions

$$\frac{\partial F}{\partial x_i} = 0 \quad , \; i = 1, \dots, n-j \tag{110}$$

But the elimination might meet with practical difficulties due to the complex, nonlinear character of F and/or h_j. The sequence of eliminations is arbitrary, and one would not automatically find the most suitable variables to eliminate.

These difficulties can be avoided by using the Lagrangian multiplier method. Consider the variations (virtual changes, infinitesimal, kinematically permissible, but not actual):

$$\delta F = \frac{\partial F}{\partial x_1}\delta x_1 + \dots + \frac{\partial F}{\partial x_n}\delta x_n = 0 \tag{107a}$$

and

$$\delta h_k = \frac{\partial h_k}{\partial x_1}\delta x_1 + \dots + \frac{\partial h_k}{\partial x_n}\delta x_n = 0 \tag{109a}$$

for $1 \leq k \leq j$

Multiplying each condition (109a) by an arbitrary constant, the undetermined multiplier, λ_k, and adding terms that are zero individually we get

$$\frac{\partial F}{\partial x_1}\delta x_1+\dots+\frac{\partial F}{\partial x_n}\delta x_n+\sum_{k=1}^{j}\lambda_k\left(\frac{\partial h_k}{\partial x_1}\delta x_1+\dots+\frac{\partial h_k}{\partial x_n}\delta x_n\right) \tag{111}$$

or, more concisely

$$\sum_{p=1}^{n}\left(\frac{\partial F}{\partial x_p}+\sum_{k=1}^{j}\lambda_k\frac{\partial h_k}{\partial x_p}\right)\delta x_p = 0 \tag{112}$$

We have introduced j additional unknowns, λ_k. They may now be chosen so that the expression in parentheses shall vanish, for example for the last j values of p:

$$\frac{\partial F}{\partial x_p}+\sum_{k=1}^{j}\lambda_k\frac{\partial h_k}{\partial x_p} = 0 \quad \text{for } p=n-j+1,\dots,n \tag{113}$$

All remaining $\delta x_p,\ p=1,\dots,n-j$ can be chosen arbitrarily (free variation problem), so that their coefficients must vanish.

$$\frac{\partial F}{\partial x_p}+\sum_{k=1}^{j}\lambda_k\frac{\partial h_k}{\partial x_p} = 0 \quad \text{for } p=1,\dots,n-j$$

Combining (113) and (114)

$$\frac{\partial F}{\partial x_p}+\sum_{k=1}^{j}\lambda_k\frac{\partial h_k}{\partial x_p} = 0 \quad \text{for } p=1,\dots,n \tag{115}$$

The foregoing derivation may be summarized in a nutshell for the case of a single constraint, h = 0. Instead of solving the variational problem

$$\delta F = 0$$

with the constraint h = 0, we solve the free variational problem for $\bar{F} = F + \lambda h$

$$\delta\bar{F}=\delta(F+\lambda h)=\delta F+\lambda\delta h+\underbrace{\cancel{h}\,\delta\lambda}_{0,\ \text{since } h=0}=\delta F+\lambda\delta h=0 \tag{116}$$

For the practical solution, we have to deal with j + n unknowns, namely $\lambda_1, \ldots, \lambda_j, x_1, \ldots, x_n$ These are found from a system of equations corresponding to the conditions

$$\begin{aligned} h_k &= 0, \quad \text{for } k = 1, \ldots, j \\ \frac{\partial F}{\partial x_p} &= 0, \quad \text{for } p = 1, \ldots, n \end{aligned} \qquad (117)$$

It may seem as if we have complicated matters by increasing the number of unkonwns from n to j + n. But we have transformed the problem into a free variational problem, i.e., one without constraints, and where conventional equation solving techniques fail we can now apply a variety of methods designed specifically for unconstrained optimization problems.

3.32 Random search.

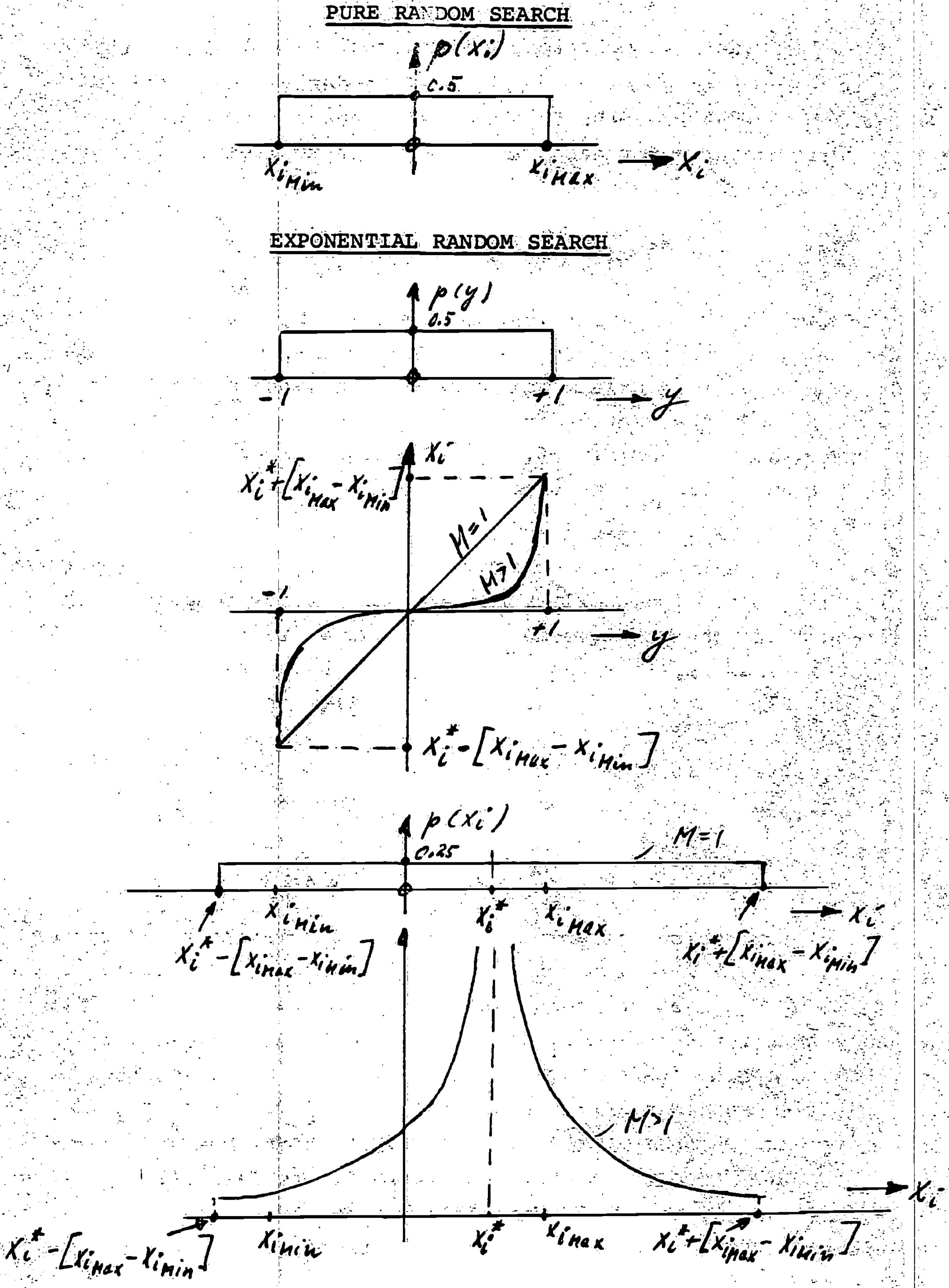
A method that has gained much popularity in recent years because of its universal applicability and suitability for computer work is the random search technique and its several variants, Refs. 76, 77, 83. Successful applications to ship design have also been reported, Refs. 60, 61, and 36 (Stephans). An illustration of the idea is given in Fig. 20.

Let us first consider the basic form of the method, the pure random search. In seeking an extremum of the object function

$$F = F(x_1, \ldots, x_i, \ldots, x_n)$$

we treat each design variable, x_i, as a random variable. Moreover, we specify equal likelihood for each value of x_i in the feasible range $x_{i_{min}}$ to $x_{i_{max}}$ given for this variable. That is, the probability density function for each x_i is uniform as shown in Fig. 21.

Values of the design variables are obtained randomly, that is usually from random number tables, and the object function F is evaluated for each new combination. Only the best value of F and its associated x-vector are saved, unsuccessful trials are forgotten. This sampling process continues until a specified number of trial points has been taken.



PURE RANDOM SEARCH

EXPONENTIAL RANDOM SEARCH

Fig. 21: Probability density and transformation functions. Modified from Ref. 60.

Brooks and Carnahan, Refs. 83, 76, 77, have examined the efficiency of this search process. They derive the number of random points p required to approximate the extremum of F within a given tolerance for each random variable x_i (see below), and with specified probability s:

$$p = \frac{\ln(1-s)}{\ln(1-b)} \approx - \frac{\ln(1-s)}{b} \qquad (118)$$

where $$b = \prod_{i=1}^{n} a_i = \prod_{i=1}^{n} \frac{\alpha_i}{l_i}$$

and $$l_i = x_{i_{Max}} - x_{i_{Min}}$$

= given search interval of x_i

α_i = desired tolerance, or interval of uncertainty for x_i at the end of the search

a_i = fractional tolerance for x_i

b = product of all fractional tolerances

As an illustration, Table III gives random point requirements for s = 0.9.

Table III

Required number of points, p

b	n = 1	n = 2	n = 3	n = 4	n = 5
0.1	22	230	$2.3 \cdot 10^3$	$2.3 \cdot 10^4$	$2.3 \cdot 10^5$
0.01	230	23035	$2.3 \cdot 10^6$	$2.3 \cdot 10^8$	$2.3 \cdot 10^{10}$
0.001	2301	$2.3 \cdot 10^6$	$2.3 \cdot 10^9$	$2.3 \cdot 10^{12}$	$2.3 \cdot 10^{15}$

Evidently, the effort gets excessive rapidly as the number of search dimension increases. Eq. 118 demonstrates that the growth with n is roughly exponential.

One basic weakness of the method is that it does not learn from earlier trials. On the other hand, it is basically the same feature, namely the randomness of every new trial, that ensures the success of the method in optimizing multimodal functions where many more efficient methods fail.

But it is fortunately possible to incorporate some learning in the random search, thus greatly increasing the efficiency while conserving practically the same degree of universality. The capability of learning turns the random search from a simultaneous into a sequential procedure.

The imbedded random search incorporates the following learning feature. After a certain number of trials the search region is reduced to a certain neighborhood of the best point found up to that time. This may be repeated several times. For more details, see Refs. 76, 77.

In the exponential random search methods, an analogous feature is that toward the later stages of the process the search intensity is gradually increased around the best point encountered up to that time. Ref. 60 gives an algorithmic description of the method of which the following is a summary.

Every random variable, x_i, is related to the random variable y by means of an exponential transformation

$$x_i = x_i^* + \left[x_{i_{Max}} - x_{i_{Min}}\right] y^M \qquad (119)$$

where x_i^* = the value of x_i associated with the best result for F at any intermediate stage of the search

$x_{i_{Max}}, x_{i_{Min}}$ = limits of the specified search interval

M = exponent, real number.

The random variable y has a uniform probability density $p(y) = 0.5$ in the interval $-1 \leq y \leq +1$, see Fig. 21, and $p(y) = 0$ for any other y.

The relation (119) will transform the element $p(y)dy$ of the sample into a corresponding element of x_i, $p(x_i)dx_i = p(y)dy$, hence

$$p(x_i) = \frac{p(y)}{\frac{dx_i}{dy}} = \frac{0.5}{\frac{dx_i}{dy}} = \frac{0.5}{M\left[x_{i_{Max}} - x_{i_{Min}}\right]} \cdot \left\{ \frac{x_i - x_i^*}{x_{i_{Max}} - x_{i_{Min}}} \right\}^{\frac{1-M}{M}} \qquad (120)$$

The purpose of this transformation is to increase the probability density near x_i* step by step by corresponding increase of M. For $M = 1$, $p(x_i)$ is uniform, see Fig. 21, and we have the situation of the pure random search. Note, however, the extended range of x_i from $x_i* - [x_{iMax} - x_{iMin}]$ to $x_i* + [x_{iMax} - x_{iMin}]$, which is necessary in order to safely accommodate the original search interval whenever x_i* is near one boundary. The probability density is consequently only 0.25, and fifty percent of all trial points will fall outside the given interval. They simply have to be discarded.

As M is increased the search intensifies in a narrowing interval of x_i centered around x_i*, Fig. 21. The variation of M has to be done cautiously, particularly where several extreme points occur. Mandel and Leopold, Ref. 60, obtained good results with

k/K	M
0 - 0.6	1
0.6 - 0.8	3
0.8 - 0.9	5
0.9 - 1.0	7

K = total number of trials

k = counting subscript, $1 \leq k \leq K$.

The exponential random search is far more efficient than the pure random search. But the effort still grows very rapidly with the number of search dimensions.

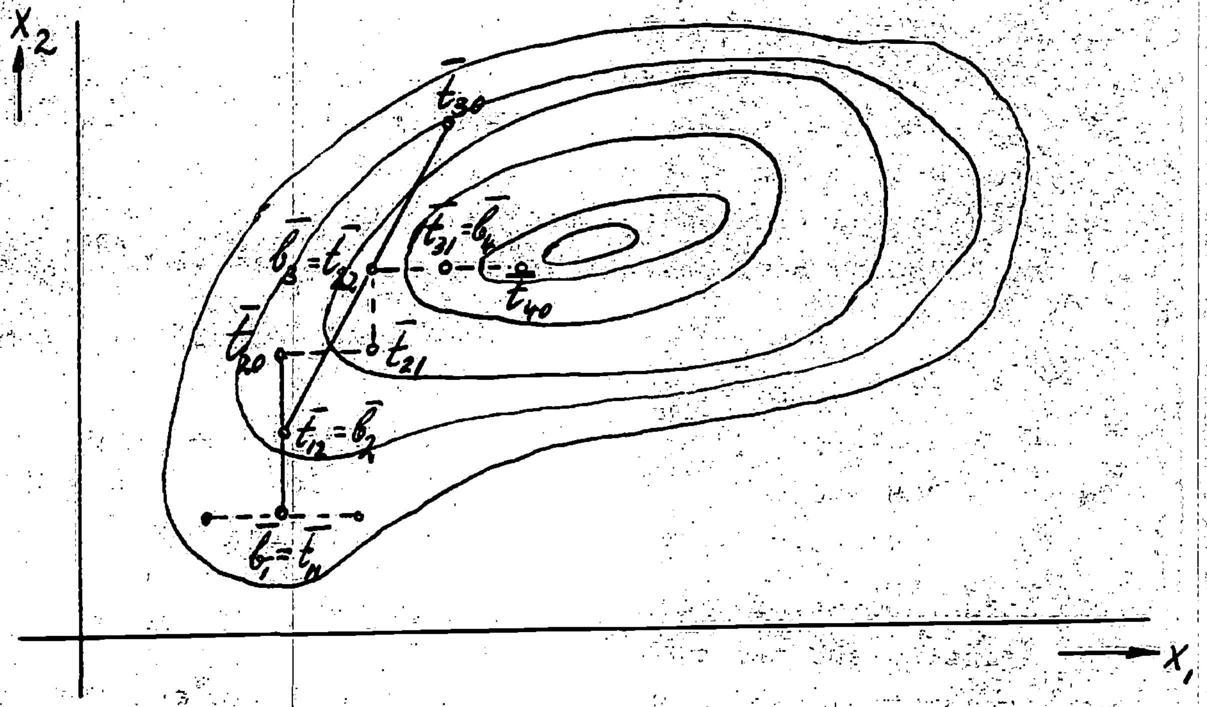
3.33 Direct search.

The label "Direct search" has been attached to a variety of unconstrained optimization methods, all taking only function values as input, and all designed to get around "ridge" problems other methods are suffering from. We want to concentrate on a version originated by Hooke and Jeeves, Ref. 84. This version, and its numerous successors, are characterized by a two-step sequence: An exploratory local "pattern search" alternates with a global "pattern move." For another variant of "direct search," which we would prefer to classify as rotating coordinate search, see Ref. 85.

The direct search by Hooke and Jeeves is well described in Ref. 77. The following must be limited to brief, somewhat modified excerpts from this source.

DESCRIPTION OF ALGORITHM

Example sketched for two decision variables x_1, x_2.



STEPS:

1. Select arbitrary starting point $\bar{b}_1$.

LOCAL PATTERN SEARCH:

2. Sample function at $\bar{b}_1 \pm \bar{\delta}_1$, and $\bar{b}_1$, where

$$\bar{\delta}_1 = \begin{bmatrix}\delta_1\\ 0\\ \cdots\end{bmatrix} \qquad \delta_1 = \text{assumed step width.}$$

Keep best point $\bar{t}_{11}$.

3. Sample function at $\bar{t}_{11} \pm \bar{\delta}_2$, in addition to $\bar{t}_{11}$, where $\bar{\delta}_2 = \begin{bmatrix}\delta_2\\ \cdots\end{bmatrix}$ δ_2 = assumed step width. Keep best point $\bar{t}_{12}$.

4. Continue until all decision variables have been sampled. In our example the final point is $\bar{t}_{12}$. This becomes the new base point $\bar{b}_2$.

GLOBAL PATTERN MOVE:

5. Go in the direction of $\bar{b}_2 - \bar{b}_1$, or generally $\bar{b}_{k+1} - \bar{b}_k$, adding same step. A new temporary point $\bar{t}_{20}$ is obtained ($\bar{t}_{k+1,0}$).

$$\bar{t}_{20} = \bar{b}_1 + 2(\bar{b}_2 - \bar{b}_1) = \bar{b}_2 + (\bar{b}_2 - \bar{b}_1)$$

or

$$\bar{t}_{k+1,0} = \bar{b}_k + 2(\bar{b}_{k+1} - \bar{b}_k) = \bar{b}_{k+1} + (\bar{b}_{k+1} - \bar{b}_k)$$

6. If the function value at $\bar{t}_{20}$ ($\bar{t}_{k+1,0}$) is an improvement over that at $\bar{b}_2$ ($\bar{b}_{k+1}$), use $\bar{t}_{20}$ as new base point for a local search, step 2. The terminal point of this local search becomes $\bar{b}_3$ ($\bar{b}_{k+2}$), and a global move $\bar{b}_2 \to \bar{b}_3$ ($\bar{b}_{k+1} \to \bar{b}_{k+2}$) follows, and so on, alternatingly.

   If after a pattern move the function value at $\bar{t}_{k+1,0}$ is not an improvement over that at $\bar{b}_{k+1}$, $\bar{b}_k$ is dropped from consideration, $\bar{b}_{k+1}$ is used as new base point and we go back to step 2, local search.

CLOSING CONDITIONS:

7. If at the end of any local pattern search

$$\bar{b}_2 = \bar{b}_1,$$

   reduce step widths $\delta_1, \delta_2, \ldots$ and try again.

8. If the step widths have become less than the specified permissible errors

$$\delta_1 \leq \varepsilon_1$$
$$\delta_2 \leq \varepsilon_2$$
$$\ldots\ldots$$

   terminate search.

3.34 A general nonlinear programming method.

Kavlie, Kowalik, Lund, and Moe, in 1966, Ref. 86, presented a promising method, designed to deal with the general nonlinear programming problem as formulated in Eqs. (107) through (109). Ref. 86 refers to earlier fundamental work by Kowalik, Ref. 88. Refs.

86 and 87 give applications of the method in ship structural design.

Given the problem of optimizing the object function $F(x)$ under inequality and equality constraints, $g_i(x) \geq 0$ and $h_j = 0$, respectively. The method defines a new object function

$$\bar{F}(x, r_k) = F(x) + r_k \sum_{i=1}^{m} \frac{1}{g_i(x)} + \frac{1}{\sqrt{r_k}} \sum_{j=1}^{t} h_j^2(x) \qquad (121)$$

$j < n$

x = n - component vector of design variables

r_k = a parameter, approaching zero in successive approximation steps

This transformation converts the constrained optimization problem into an unconstrained one. For, heavy penalties are placed by the r_k-terms on violating the equality constraints, $h_j = 0$, and on being right at the boundary, $g_i = 0$. The latter is not necessarily desired because the optimum generally lies directly on some governing inequality constraint. But the influence of the second term is limited to a small neighborhood of the lines $g_i = 0$, and by letting r_k approach zero one can get arbitrarily close to the true optimum.

The big advantage of the transformation is that any of the efficient search methods for unconstrained variables may now be applied to the problem. The authors of Refs. 86, 87 use Powell's direct search (or rotating coordinate search).

Special precautions are necessary to find an initial estimate in the feasible domain. After that, with some caution in step width and/or checks of initial constraints, one will converge to the optimum from the feasible side of the boundary.

A simple, illustrative example is given in Ref. 86 in the structural optimization of a box girder design.

3.35 Dynamic programming.

Dynamic programming is not an optimization technique as such, but an approach to optimization, relying on some actual optimization techniques of any desired type for the steps involved.

Dynamic programming is designed for the optimization of systems that have a stagewise structure. Consider, for example, the system of Fig. 22. It has three stages whose returns R_1, R_2, R_3 are additive. The measure of merit function $F = R_1 + R_2 + R_3$ is to be maximized. (Dynamic programming is not limited to additive stage returns, but this is its most frequent application, see Ref. 82.)

The stage returns depend on the decision variables, D_1, D_2, D_3, in an arbitrary manner, for example as shown in Fig. 23, and they also depend on the state variables, X_1, X_2, X_3, passed on from stage to stage.

$$\begin{aligned} R_1 &= R_1(X_1, D_1) \\ R_2 &= R_2(X_2, D_2) \\ R_3 &= R_3(X_3, D_3) \end{aligned} \qquad (122)$$

These relationships must be known, and we must further have equations relating stage inputs and outputs

$$\begin{aligned} X_2 &= t_3(X_3, D_3) \\ X_1 &= t_2(X_2, D_2) \\ X_0 &= t_1(X_1, D_1) \end{aligned} \qquad (123)$$

Eqs. (122) and (123) constitute the mathematical model of the system.

We could, of course, accomplish an optimization of F for the system by an exhaustive variation of all decision variables. But this is a tedious process which we can avoid by exploiting the serial multistage structure of the system. In dynamic programming, we will thus obtain a sequence of stage optimizations, and by composing the results we can optimize the system not in the trivial sense of stage optimality, but in the sense of overall optimality.

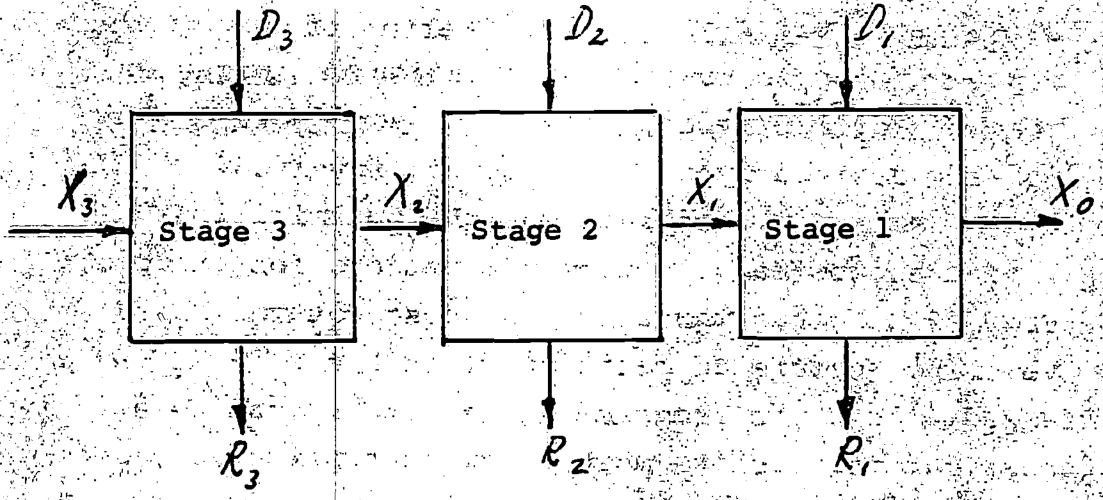


Fig. 22: Three-stage system

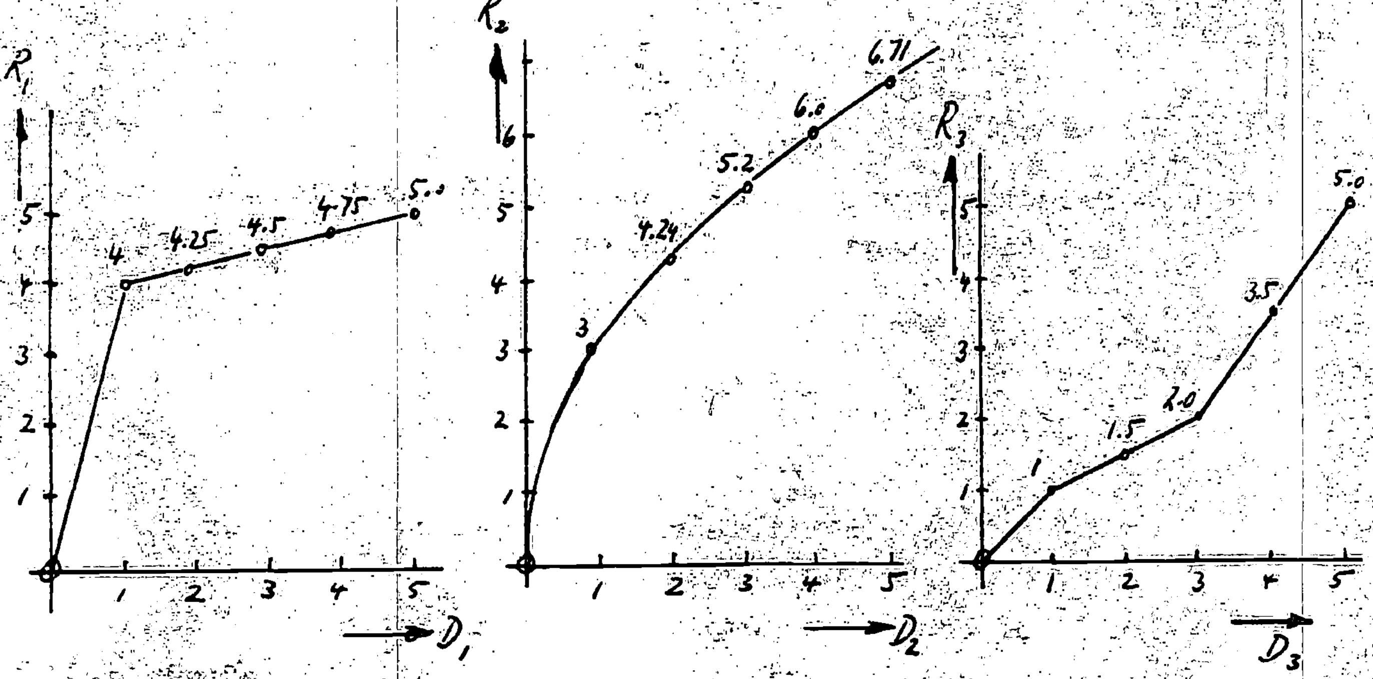


Fig. 23: Stage return functions

Note, first of all, that due to the serial nature of the system the state variables depend only on the decisions prior to their stage, and hence the analogous is true for the stage returns.

$$\begin{aligned} & & R_3 &= R_3(X_3, D_3) \\ X_2 &= t_3(X_3, D_3) & R_2 &= R_2(X_2, D_2) = R_2(X_3, D_3, D_2) \\ X_1 &= t_2(X_2, D_2) = t_2(X_3, D_3, D_2) & R_1 &= R_1(X_1, D_1) = R_1(X_3, D_3, D_2, D_1) \\ X_0 &= t_1(X_1, D_1) = t_1(X_3, D_3, D_2, D_1) & & \end{aligned} \tag{124}$$

We are interested in a maximum (minimum) of the total return F.

$$F(X_3) = \max_{D_3, D_2, D_1} \left\{ R_3(X_3, D_3) + R_2(X_3, D_3, D_2) + R_1(X_3, D_3, D_2, D_1) \right\} \tag{125}$$

To transform this into a series of stage maximizations, we make use of the simple theorem that for any real-valued functions $f_1(x_1)$ and $f_2(x_1, x_2)$

$$\max_{x_1, x_2} \left[f_1(x_1) + f_2(x_1, x_2) \right] = \max_{x_1} \left[f_1(x_1) + \max_{x_2} f_2(x_1, x_2) \right] \tag{126}$$

This may be applied repeatedly which leads to

$$F(X_3) = \max_{D_3} \left[R_3(X_3, D_3) + \max_{D_2} \left\{ R_2(X_2, D_2) + \max_{D_1} R_1(X_1, D_1) \right\} \right] \tag{127}$$

The principle underlying this derivation has also been formulated as the <u>principle of optimality</u> of dynamic programming (Bellman), Ref. 82: "An optimal set of decisions has the property that whatever the first decision is, the remaining decisions must be optimal with respect to the outcome which results from the first decision."

For a general derivation for n-stage systems, and for not necessarily additive stage returns, as well as for extensions to nonserial systems, see also Ref. 82.

Equation (127) contains all necessary ground rules for a computational procedure.

1. Optimize stage 1 with respect to D_1 with X_1 as a parameter.
2. Express X_2 in terms of X_1, and optimize stages 1 and 2 with respect to D_2, with X_2 as a parameter.
3. Continue analogously. In the example, the final optimization is for stages 1 through 3 over D_3 with X_3 as parameter.
4. If X_3 is not fixed as input, optimize $F(X_3)$. Resubstitute to get all other optimal stage and decision variables.

The following example may serve as an illustration. For the system of Fig. 22 and the stage returns of Fig. 23, maximize $F = R_1 + R_2 + R_3$, subject to the resource constraint

$$D = D_1 + D_2 + D_3 \leq 5 \text{ units} \qquad (128)$$

The example is a nonlinear resource allocation problem where the stages might for example represent phases of a cargohandling system (container handling in yard, on board, see leg) and the question is where to best invest a limited amount of resources.

Table IV: Solution of dynamic programming problem

D	$f_1 = R_1$	$R_2(D_2)$	f_2	D_1,D_2	$R_3(D_3)$	f_3	D_1,D_2,D_3
0	0	0	0	0,0	0	0	0, 0, 0
1	4	3	4	1,0	1	4	1, 0, 0
2	4.25	4.24	7	1,1	1.5	7	1, 1, 0
3	4.5	5.20	8.24	1,2	2.0	8.24	1, 2, 0
4	4.75	6.00	9.20	1,3	3.5	9.24	1, 2, 1
5	5	6.71	10.0	1,4	5	10.20	1, 3, 1

The state variables in this problem simply express how much of the resource is left over after the corresponding stage.

$$\begin{aligned} X_0 &= X_1 - D_1 \geqslant 0 \\ X_1 &= X_2 - D_2 \\ X_2 &= X_3 - D_3 \\ X_3 &\leq 5 \end{aligned} \qquad (129)$$

The steps of the computation in Table IV correspond to the above sequence, steps 1 through 4.

1. In optimizing stage 1, the input resource, X_1, and hence the allocation to the stage, D_1, may be anything between 0 and 5. The optimization is trivial, and consists of using up all that is available so that $f_1 = R_1$, where f_n will always denote the suboptimum up to and including stage n $\left[f_n = \max_{D_n}(R_n + f_{n-1}),\ f_1 = \max_{D_1} R_1\right]$, This is because R_1 is a monotonically increasing function of D_1. X_0 is always zero, hence $D_1 = X_1$.
2. In optimizing stages 1 and 2, the available resources, D, column 1, must be split optimally. This is done so that (129) is satisfied from which

   $$D_2 = X_2 - X_1 = X_2 - D_1$$

   Column 1, D, functions as X_2 temporarily, pretending we are allocating D entirely to stages 1 and 2.

   The suboptimum f_2 and the associated best values of D_1, D_2 are listed in the table for any value of $D = X_2$. They are found by systematically trying to maximize $R_2 + f_1$, given $X_2 = D_1 + D_2$.
3. Correspondingly, in the next step one finds f_3 and the best (D_3, D_2, D_1)-combinations, varying $D = X_3$, and maximizing $R_3 + f_2$. The benefit of the stagewise decomposition becomes apparent here in that we need only consider optimal pairs of D_1, D_2 and match them with D_3, rather than all feasible combinations. The more stages the greater the savings due to eliminating nonoptimal strategies from further consideration in this manner.
4. The optimal allocation, $D_1 = 1$, $D_2 = 3$, $D_3 = 1$, results in $f_3 = 10.20$ units. All stage returns are monotonically increasing functions, and hence there was no question that we should allocate the whole available resource:

$f_{s_{Max}} = 10.2$ for $D = 5$.

The value of dynamic programming in engineering is only beginning to be appreciated. For a few more examples see Refs. 77, 82, and for a ship structural design application, 89.

Whenever in a complex, nonlinear optimization problem one has to deal with additive returns from different activities or objects one has good reason to believe that stage decomposition of the problem will be feasible and fruitful.

3.4 Examples.

3.41 The Minimum Ship Resistance Problem.

Lin, Webster, and Wehausen in their paper "Ships of Minimum Total Resistance," Ref. 90, attack one of the most fascinating problems of ship hydrodynamics. They attempt to minimize the total resistance of a ship as approximated on the basis of linearized wave theory (Michell) and of the flat plate concept of frictional resistance. The following brief excerpts may serve to illustrate how this leads to an optimization problem that may be treated by the Lagrangian multiplier method.

The hull form is generated by a continuous distribution of singularities in the rectangular centerplane,

$$\left.\begin{array}{r} -1 \le x \le 1 \\ 0 \le z \le 1 \end{array}\right\} \quad \text{nondimensional coordinate ranges}$$

The equation of the hull is represented by double Fourier series:

$$f_a(x,z) = \sum_{m=1}^{M}\sum_{p=1}^{P} a_{mp} \cdot \cos\tfrac{1}{2}(2m-1)\pi x \cdot \cos\tfrac{1}{2}(2p-1)\pi z$$

$$f_b(x,z) = \sum_{m=1}^{M}\sum_{p=1}^{P} b_{mp} \cdot \cos\tfrac{1}{2}(2m-1)\pi x \cdot \cos\tfrac{1}{2}(2p-1)\pi z \qquad (130)$$

f_a, f_b = afterbody, and forebody equation, respectively

From thin ship theory, one obtains for the wave resistance of the hull (Michell wave resistance, R_M):

$$C_M = \frac{R_M}{\rho g B^2 T} =$$

$$= \frac{1}{\pi}\frac{2T}{L}\frac{1}{\gamma_0}\int_{\gamma_0}^{\infty}(C^2+S^2)\frac{\gamma^4}{\sqrt{\gamma-\gamma_0^2}}\,d\gamma$$

$$\gamma_c = \frac{1}{2F_N^2} = \frac{gL}{2\,v_0^2}$$

$$C + iS = \int_{-1}^{+1}dx\int_{0}^{1}dz\, f(x,z)\cdot\exp\left(-\frac{2T}{L\gamma_0}\gamma z^2 + i\gamma x\right)$$

L, B, T = length, beam, draft
F_N, v_0 = Froude number, speed
$f(x,z)$ = hull equation, $f_a(x,z)$ or $f_b(x,z)$

The viscous part of the resistance, R_V, is represented simply as

$$C_V = \frac{R_V}{\rho g B^2 T} = \frac{\frac{1}{2}\rho v_0^2\, S\, C_F}{\rho g B^2 T} = \qquad (131)$$

$$= \frac{1}{2\gamma_0}\frac{L^2}{B^2}C_F\iint_{S_0}\left(1+\frac{B^2}{L^2}f_x^2+\frac{B^2}{4T^2}f_z^2\right)^{1/2}dx\,dz$$

C_F = friction coefficient from standard friction line
S = wetted surface
f_x, f_z = partial derivatives of f

Two different problems are formulated:

Problem I:

Minimize the total resistance for a given afterbody, i.e.:

Given the set of a_{mp}, optimize the set of b_{mp} for the forebody, ensuring a smooth transition at midship section. Hence, minimize $T = C_M + C_V$

$$\text{subject to} \quad \sum_{m=1}^{M} b_{mp} = \sum_{m=1}^{M} a_{mp}, \text{ for any } p=1,\ldots,P \tag{132}$$

<u>Problem II</u>:

Minimize the total resistance of the ship for a given volumetric coefficient.

Or, minimize $T = C_M + C_V$

subject to

$$C_{\nabla} = \frac{B}{T} \cdot \left(\frac{T}{L}\right)^2 \frac{2}{\pi^2} \sum_{m=1}^{M} \sum_{p=1}^{P} \frac{(-1)^{m+p}}{(2m-1)(2p-1)} b_{mp} = \tag{133}$$

$$= k_1, \quad k_1 = \text{a constant.}$$

Both problems are nonlinear optimization problems with equality constraints, hence they transform into the Lagrangian multiplier formulations for unconstrained functions $\overline{T}$:

<u>Problem I</u>: Minimize

$$\overline{T} = C_M + C_V + \sum_{p=1}^{P} \lambda_p \left(\sum_{m=1}^{M} b_{mp} - \sum_{m=1}^{M} a_{mp} \right) \tag{134}$$

<u>Problem II</u>: Minimize

$$\overline{T} = C_M + C_V + 2\lambda \left[\sum_{m=1}^{M} \sum_{p=1}^{P} \frac{(-1)^{m+p} \cdot b_{mp}}{(2m-1)(2p-1)} - \frac{T}{B}\left(\frac{L}{T}\right)^2 \frac{\pi^2}{2} \cdot C_{\nabla} \right] \tag{135}$$

The unknowns b_{nq}, and λ_p, are found from the conditions

$$\frac{\partial \overline{T}}{\partial b_{nq}} = \dots = 0, \quad n = 1, \dots, M; \; q = 1, \dots, P$$

$$\frac{\partial \overline{T}}{\partial \lambda_p} = \dots = 0, \begin{cases} p = 1, \dots, P & (\text{Problem I}) \\ p = 1 & (\text{Problem II}) \end{cases} \tag{136}$$

For the results of this study, see Ref. 90.

3.42 Pien's Formulation of the Resistance Problem.

Pien deals with the minimum resistance problem in an original and pragmatic manner, Refs. 91 to 93. Some of most important assumptions are:

*Off-centerplane singularity distributions are used. Despite some fundamental doubts in the uniqueness of the resistance expressions so obtained, Pien follows this path for pragmatic reasons in order to generate beamier and fatter hulls than would be possible by thin ship theory.

*The viscous resistance is left out of consideration except that the principal dimensions are initially chosen so as to make for small wetted surface.

*Only the forebody wave resistance is minimized, the afterbody being symmetrical or even arbitrary. This is done to keep the bow wave system small by itself, rather than trying to capitalize from bow-stern wave system interference.

*Constraints are imposed on volume, midship area, entrance angle of the waterline, and bottom flatness, or at least on some of these properties.

*Bulbous bows may be introduced and optimized separately.

Despite these numerous differences from the formulation of section 3.41, Pien is also led to a nonlinear programming problem with equality constraints. Although he chooses the elimination method to obtain a solution, one can instead easily apply Lagrangian multipliers as before. See Refs. 91, 92 for details.

3.43 Structural Optimization by Nonlinear Programming.

A simple, illustrative example for the optimization of structures as to cost and/or weight is the opening example in the paper by Moe and Lund already mentioned, Ref. 87.

A constant cross-section box girder of hinged support is under a uniform distributed load, Fig. 24. Given the web thickness of the girder, t_S, select flange thickness, t_F, and girder height, h, so as to minimize girder cross section, and thereby weight and cost.

The objective function is

$$A = 2t_S \cdot h + 2t_F \cdot b, \qquad (137)$$

and the constraints:

Maximum allowable shear stress (τ_{ALL}): $\tau \leq \tau_{ALL}$

Maximum allowable bending stress($\sigma_{B_{ALL}}$): $\sigma_B \leq \sigma_{B_{ALL}}$

Buckling stress (σ_{CRIT}): $\sigma_B \leq \sigma_{CRIT}$

Maximum deflection (δ_{ALL}): $\delta \leq \delta_{ALL}$

Constraints in standard form:

$$g_1 = t_F > 0 \qquad g_5 = \frac{\sigma_{CRIT}}{\sigma_B} - 1 \geq 0$$

$$g_2 = h > 0 \qquad g_6 = \frac{\delta_{ALL}}{\delta} - 1 \geq 0 \qquad (138)$$

$$g_3 = \frac{\tau_{ALL}}{\tau} - 1 \geq 0$$

$$g_4 = \frac{\sigma_{BALL}}{\sigma_B} - 1 \geq 0$$

Fig. 25 illustrates the optimization problem geometrically. We are interested in the minimum of A, compatible with the constraints. In the example, g_5 is the overriding constraint. Fig. 26 shows section α-α through the optimization space. It also contains the modified objective functions, $\bar{A}$, which may be minimized without regard to constraints:

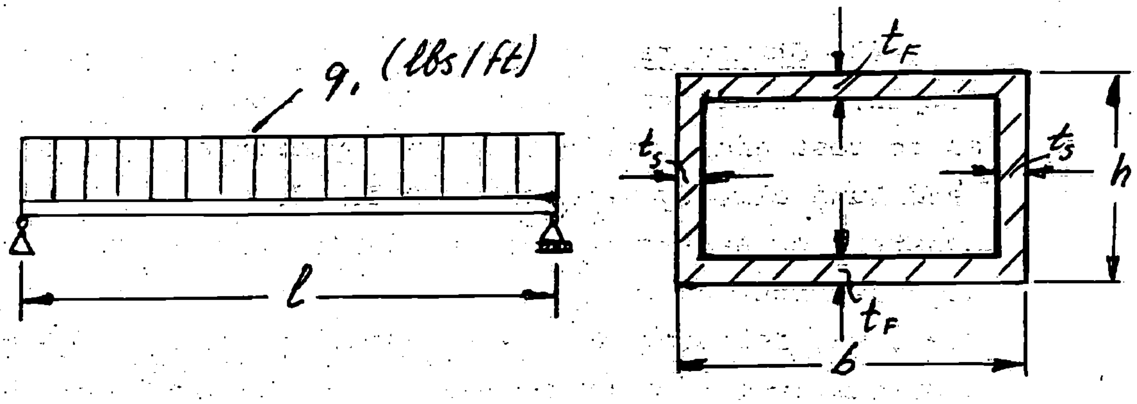


Fig. 24: Box girder (from Ref. 87)

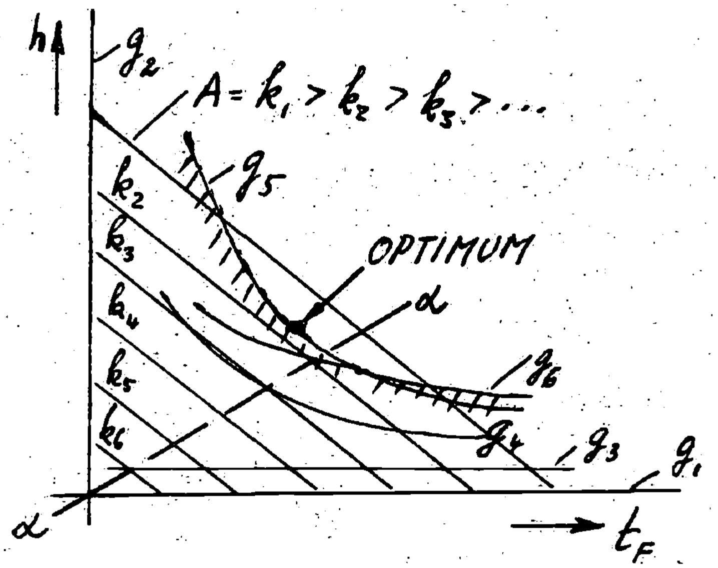


Fig. 25: Decision space (from Ref. 87)

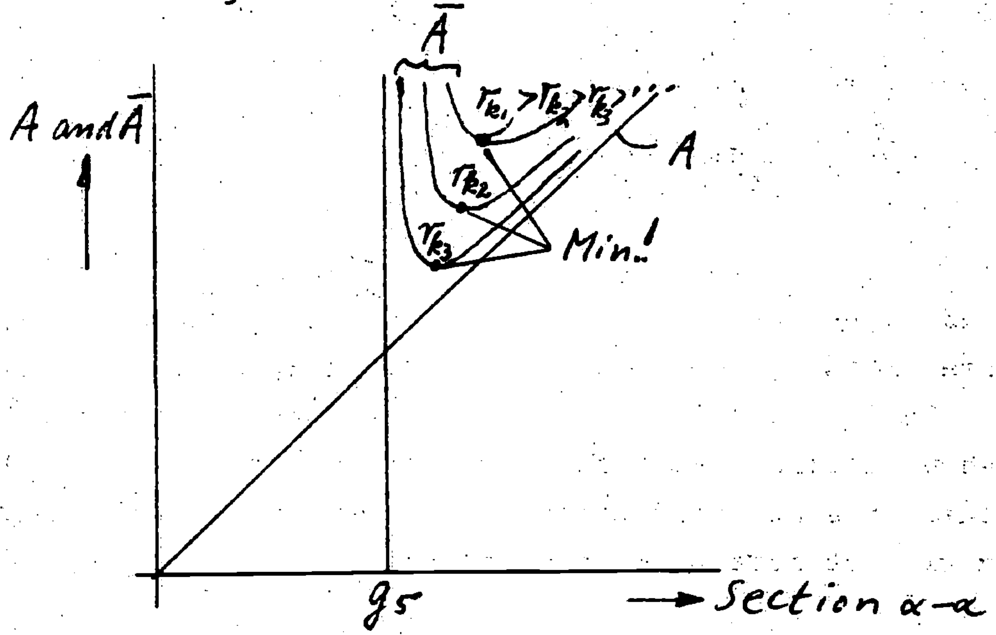

$\bar{A}$
A and $\bar{A}$
$r_{k_1} > r_{k_2} > r_{k_3} > \dots$
A
r_{k_2}
r_{k_3}
Min.!
g_5
Section α–α

Fig. 26: Original and transformed objective functions (from Ref. 87)

$$\bar{A}(t_F, h, r_k) = A(t_F, h) + r_k \sum_{i=1}^{m} \frac{1}{g_i(t_F, h)} \qquad (139)$$

The solution proceeds by seeking mimima on the surfaces $\bar{A}$ for successively smaller values of r_k, using a special variant of the direct search technique. For numerical values and further comments, see Ref. 87.

3.44 <u>Miscellaneous other applications</u>.

Other applications of optimization methods to ship design include:

*The exponential random search studies by Mandel and Leopold, Ref. 60, as mentioned in sections 2.35, and 3.32.

*A linear programming formulation of some special cargo handling questions by Krappinger, Ref. 94.

*Other applications of the nonlinear programming method of section 3.43 to structural optimization, Refs. 86, and 88.

*An application of dynamic programming to plate dimensioning in ship structures by Moses and Tønnesen, Ref. 89.

3.5 References 75 through 94.

75. Lanczos, C., "The Variational Principles of Mechanics," University of Toronto Press, Toronto, 1960.

76. Katz, D. L., Carnahan, B., et al., "Computers in Engineering Design Education," Volume I, Summary Report, The University of Michigan, College of Engineering, April 1966.

77. Carnahan, B., Wilkes, J. O., "Numerical Methods, Optimization Techniques, and Simulation for Engineers," Lecture Notes, The University of Michigan, Engineering Summer Conferences, May 1968.

78. Carnahan, B., Wilkes, J. O., "Algorithms and Numerical Methods," The University of Michigan, Ann Arbor, 1968.

79. Wilde, D. J., "Optimum Seeking Methods," Prentice-Hall, Englewood Cliffs, New Jersey, 1964.

80. Gass, S. I., "Linear Programming," 2nd ed., McGraw-Hill Book Company, New York, 1964.

81. Kuenzi, H. P., Krelle, W., "Nonlinear Programming," Blaisdell Publishing Company, New York, 1966.

82. Nemhauser, G. L., "Introduction to Dynamic Programming," John Wiley and Sons, Inc., New York, 1966.

83. Brooks, S. H., "A Discussion of Random Methods for Seeking Maxima," Operations Research, Vol. 6, March 1958.

84. Hooke, R., and Jeeves, T. A., "Direct Search Solution of Numerical and Statistical Problems," Journal of the Association for Computing Machines, Vol. 8, April 1962.

85. Powell, M. J. D., "An Efficient Method for Finding the Minimum of a Function of Several Variables Without Calculating Derivatives," Computer Journal, 1964.

86. Kavlie, D., Kowalik, J., Lund, S., and Moe, J., "Design Optimization Using a General Nonlinear Programming Method," European Shipbuilding, No. 4, 1966.

87. Moe, J., Lund, S., "Cost and Weight Minimization of Structures with Special Emphasis on Longitudinal Strength Members of Tankers," De Ingenieur, No. 47 and 49, 1967, also reproduced in Ref. 36.

88. Kowalik, J., "Nonlinear Programming Procedures and Design Optimization," Acta Polytechnica Scandinavica, No. Ma13, Trondheim, 1966.

89. Moses, F. Tönnesen, A., "Dynamic Programming for Computing Optimal Plate Dimensions in Some Ship Structures," European Shipbuilding, 1967.

90. Lin, W.-C., Webster, B. C., Wehausen, J. C., "Ships of Minimal Total Resistance," Proceedings of the International Seminar on Theoretical Wave-Resistance, volume III, The University of Michigan, Ann Arbor, 1963.

91. Pien, P. C., Moore, W. L., "Theoretical and Experimental Study of Wave-Making Resistance of Ships," Proceedings of the International Seminar on Theoretical Wave-Resistance, volume I, The University of Michigan, Ann Arbor, 1963.

92. Pien, P. C., "The Application of Wavemaking Resistance Theory to the Design of Ship Hulls with Low Total Resistance," Fifth Symposium on Naval Hydrodynamics, Bergen, Norway, September 1964.

93. Pien, P. C., "Some Experimental Results of Hull Form Research," DTMB-Report 2144, Washington, D.C., November 1965.

94. Krappinger, O., "Determination of Most Advantageous Cargohandling Gear for Two Cargo Holds," internal report, in German, Institut für Schiffbau, University of Hamburg, Germany, 1965.

4. Computer-Aided Detail Design.

4.1 Introduction.

Detail design is the link between preliminary (or concept) design and fabrication. The input to the ship detail design phase corresponds by and large to the contractual definition of the ship (contract, specifications and plans: Lines, general arrangements, midship section, some other plans). The output is normally a set of working plans, a bill of materials, etc., in sufficient detail for the production of the ship.

There is, of course, less freedom in detailing than in the earlier design stages. The emphasis is on compliance with contract, specifications, and rules, validation of proposed scantlings, avoidance of conflicts among competing systems, ease of fabrication, standardization, and, where any choice is left, minimization of cost and weight. None of the steps involved is basically very difficult for a detailing engineer with some experience and access to all the drawings and data files being produced in the course of the design and detailing process.

The attempt to computerize ship detailing is nevertheless a task of great magnitude, not the mathematics, but the logical organization. The volume of input and output is enormous, and the organization of an efficient data filing system requires much ingenuity.

Because of the scope of these developments it would not be economical to computerize ship detailing only halfway, say, for 50 percent of all parts. The data bank and logic developed should be exploited throughout the detailing process. This may explain why only so few detailing systems are under development or in operation.

It should be noted that computer assistance in detailing should be conprehensive, true, but the engineer should stay in control over the decision process (sequence and content), since complete automation seems neither feasible nor desirable. Current developments in detailing systems are therefore governed by the philosophy of man-computer interaction. CASDOS and AUTOKON, two specific examples, will be taken up in sections 4.3 and 4.4.

4.2 Midship section design.

Although a preliminary midship section is usually part of the input to ship detailing, the design of the midship section is reviewed here because in the course of detailing the first design is verified and, if possible, improved.

4.21 Evans and Khoushy.

References: 95, 96.

Motivation: Parametric studies of the effect of ship principal dimensions on weight and construction cost.

Approach: Load assumptions inferred from ABS rules; scantlings determined from load assumptions for given basic configuration; systematic variation of ship length, beam, draft, depth. Variation of frame spacing. Design of one hold, using 4 pillars.

Examples: Transversely framed cargo ship in Ref. 95, tanker, destroyer, other ships mentioned in Ref. 96.

Measure of merit: Least weight, and least construction cost.

4.22 Buxton.

Reference: 98.

Motivation: Optimization of midship section within Lloyd's Register's rules.

Approach: LR rules computerized for local structure. The computer program evaluates a design for given principal dimensions and type of structure. Local suboptimization (girders) provided. Repeated application leads to systematic variation of principal dimensions, block coefficient, bulkhead spacing, stiffener spacing, web spacing, center tank width.

Example: Tanker, primarily midship cargo tank.

Measure of merit: Weight.

Results: Parametric weight data on midship section structural parts for preliminary design purposes.

4.23 Chapman.

Reference: 97.

Motivation: Midship section design by systematic variation of the proportions of the midship section.

Measure of merit: Weight.

4.24 Moe and Lund.

Reference: 87.

Motivation: Midship section design optimization from load assumptions for given general topology.

Approach: Nonlinear programming formulation of design problem. Design variables: Spacing of longitudinals in deck and bottom, and along sides and bulkheads, principal dimensions of ship. Constraints: Allowable stresses, etc.

Example: Tanker.

Measure of merit: Weight and/or construction cost.

4.3 CASDOS.

CASDOS (Computer-Aided Structural Detailing of Ships) is a comprehensive detailing system under development by Arthur D. Little, Inc. for the U.S. Navy. The project was initiated in 1965, and in its first phase through 1968 has amply demonstrated its feasibility. It is now in its second phase during which it will be adapted to computer systems of most recent vintage. It is contemplated to begin trial operation of the system in the Puget Sound Shipyard in the summer of 1969.

An excellent description and evaluation of the system is given in Ref. 99. Other good references are 100 through 102. The following are a few highlights.

Purpose: CASDOS is to produce structural working plans and other production inputs from given contract plans and contract stage specifications.

Input: Contract plan information (Scantlings, lines, gross geometry of structure, arrangements), rules, yard constraints (fabrication practices, attachment details, maximum plate size, tolerances, etc.). This information, after careful computer-assisted verification, is kept in the initial master file.

Philosophy: The intent is to program existing procedures. No attempt to include structural optimization analysis. CASDOS is envisioned as a growing, ever-improving system as technology advances. The system is designed for operation in batch so that the user can call upon detailing steps in any desired and meaningful sequence, and review the intermediate results carefully before the next step. Human control rather than full automation is the goal.

Calculations: Control cards initiate a detailing run for some part of the structure. The steps involved are

*A geometry run, to return all the lofting information.

*Detailing of structure (Scantling verification, connections, straking, stiffener locationing, cutouts, edge preparation, and many other details).

Output: Drawings addressed to specific assembly step; one format for every assembly operation; drawings serve as "graphic construction aids"; much lofting information may be suppressed since plate preparation is automated. CASDOS furnishes APT program and NC tape for every plate to be handled by flame-cutting. Other outputs: Bill of materials, centers of gravity.

Computer organization: See Ref. 101, and, for a few highlights, section 6.23.

4.4 AUTOKON.

AUTOKON is a system of computer programs supported by special hardware designed by the Central Institute for Industrial Research in Oslo, and the two shipyards of the Akers group in Bergen and Oslo. The system has been in successful operation in many shipyards throughout the world for more than five years. Its main capabilities are:

*Hull definition (fairing)

*Part contour definition (detailing)

*Plate nesting

The equipment to be used with the system consists of a director for numerically controlled flame-cutting equipment (ESSI director), a drafting machine for part verification (Kongsberg Kingmatic), and a nesting table.

The system is well documented in Refs. 103 through 106. The main body of Hysing's paper, Ref. 103, is enclosed in the following part of these lecture notes. A few additional comments on the principle of the fairing method may be in place.

The method takes a given set of offset points, say about 400, as input. The boundary lines are faired first and kept fixed thereafter. Then waterlines and sections are faired alternatingly until sufficient agreement is reached.

The mathematical fairing tool is a spline whose cross section varies in steps at the points of support.

The fairing criterion is a combination of two requirements:

*<u>Fairness</u> is to be ensured by minimizing the strain energy of flexure in the spline (exact curvature expression):

$$W_1 = \int_L EI \left\{ \frac{y''}{(1+y'^2)^{3/2}} \right\}^2 \sqrt{1+y'^2}\, dx \qquad (140)$$

*<u>Accuracy</u> demands minimizing the absolute value of the deviation between curve and offset, Fig. 27. However, to filter out the effect of wild points, a transformation, F, is applied to the errors, see Fig. 28.
This results in:

$$W_2 = \sum_i w_i \cdot F\left\{ \left| \underbrace{y'(x_i)\cdot\left[y_i - y(x_i)\right]}_{\varepsilon} \right| \right\} \qquad (141)$$

w_i = arbitrary weighting factor, expressing significance of the point

y_i, $y(x_i)$ = given and faired point at x_i

The combined measure of merit, to be minimized, is then

$$W = W_1 + \lambda W_2 \qquad (142)$$

λ = switch to provide proper emphasis on fairness or accuracy.

The solution proceeds in analogy to several spline methods discussed in section 1.31. The resulting hull definition is then approximated by segments of piecewise constant curvature (circular

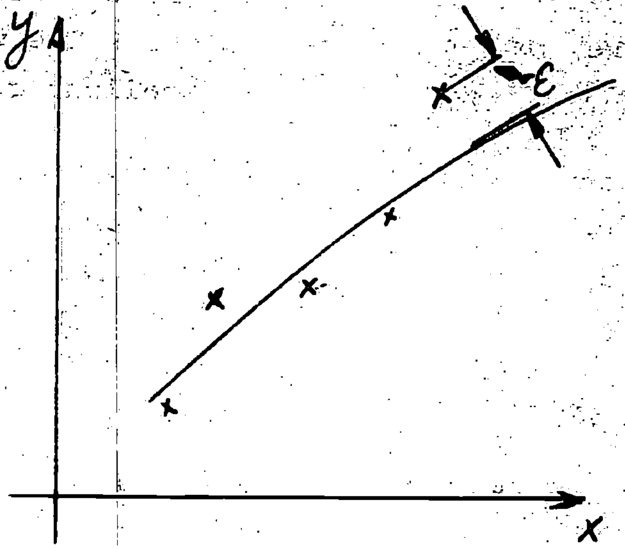


Fig. 27: Curve, offsets, and error definition

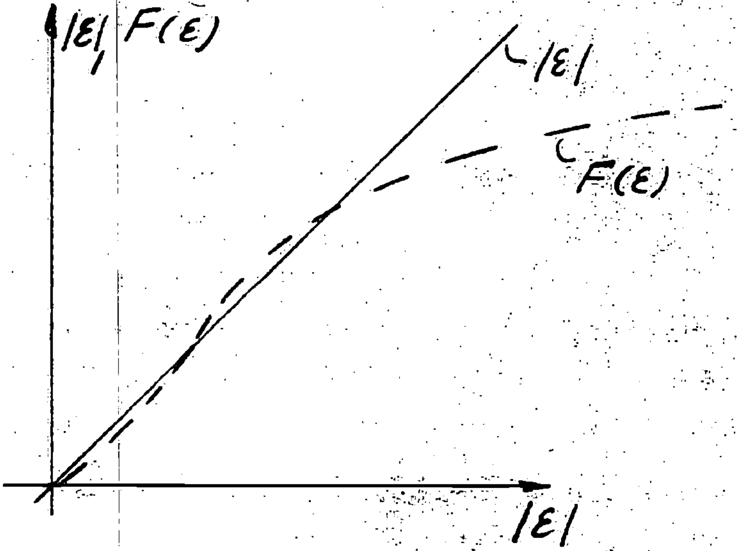


Fig. 28: Error transformation

segments) so as to facilitate use of plotters and numerical control hardware. Segment size is chosen carefully so that the difference between curve and circular segment is negligible. Slopes at the segment transitions are, of course, continuous. The whole approach is very appealing, and the results are of excellent quality.

The following notes on the Autokon system were obtained through the courtesy of Baldwin Kongsberg Company, Cincinnati, Ohio. The author is Mr. T. Hysing of CIIR, Oslo.

FROM BASIC DESIGN TO FLAMECUTTING

1. General description of the Autokon System

The shipbuilding industry is working under rather difficult conditions. Throughout the world shipyards are struggling hard to meet the requirements from the shipowners to deliver bigger, better and more efficient ships. Because of the strong competition both prices and delivery times have steadily been reduced despite the higher quality of the ships being produced.

To meet this challenge the industry has had to enforce greater modernization schemes in the shipyards where new production equipment and manufacturing methods are being introduced that have been developed during the last few decades. Of all the methods now being utilized more and more, the numerical methods may perhaps have the greatest impact on the shipbuilding industry. Today, numerical methods are being used in production among other things to direct and control important manufacturing processes such as flame cutting of plate parts. Furthermore, digital computers and electronic data processing are tools that are being used at an ever increasing rate for the design of ships' hulls, and their individual parts, for design calculations, for production planning, administrative purposes etc.

It is of interest to note the great variety of applications of these numerical methods. But even more interesting is the fact that despite their wide range, it is possible to integrate all of these activities into one system which will monitor the greater majority of the activities involved in the design and production of ships. This will make possible a marked increase in productivity, and result in greater savings both in time and manpower.

Such a completely integrated system has not yet been developed for the ship-building industry. Work is, however, carried out along these lines at many yards and institutions in several countries. The systems evolving from this work are steadily reaching a higher degree of sophistication.

The Autokon System is a result of such an attempt to build an integrated system for the design and manufacture of ships' hulls. The development has been carried out by the Central Institute for Industrial Research in Oslo, Norway, in close co-operation with two Norwegian ship yards: A/S Akers Mek. Verksted and A/S Bergens Mek. Verksteder.

This paper gives a general outline of the Autokon System. The system has been used by Norwegian ship yards for more than four years. It is also gaining acceptance among other yards which are now making an increasing use of the system.

The aim of this part of Autokon is to provide a system which carries out all the tasks involved in the production of the various types of ship plates. The system is realized by combining computers and numerically controlled flame cutters now available on the market. A set of computer programmes, especially developed for this purpose, together with suitable work routines form the basis for the realization of the system.

In its present version the Autokon System has the following merits:

i) ability to perform the fairing of hull form and to generate the corresponding table of offsets and a suitable numerical description of the body plan.

ii) ability to accept the shipbuilder's spesifications of standard shapes and of the various ship details in a simple and easy code.

iii) general facility for storing information concerning the hull and its individual parts and for referencing this information when required.

iv) facility to produce input tapes for automatic drawing of hull form and details and for numerically controlled flame cutting of the various parts.

v) facility for the nesting of ship plates on steel formats to secure an efficient utilization of material.

vi) possibilities for further expansion by including other design and manufacturing activities.

In the practical realization of the system, the ESSI numerical control system is used for the control of drawing machines or flame cutting machines. The reason for this is that the development of Autokon has been carried out as a direct continuation of the development of the ESSI system. There is, however, nothing to prevent the adaptation of other N/C-systems. In general, this requires only a change of the post processor programme. This will for most of the N/C-systems on the market require only a rather limited effort in extra computer programming.

The accompanying diagram (fig. 1) outlines the general structure of the Autokon System. The computer is employed for the transformation of the designers' ideas into ESSI director input tapes which may be used either for the automatic drawing or flame cutting of the ship details.

As indicated in the diagram the computer is fed with information about hull form, standard contour shapes, etc. as soon as these are determined during the design stage. The data are stored for later reference when needed for the generation of part contours and their accompanying control tapes.

The preparation of these data is made by various computer programmes of which the programme for numerical hull fairing is the most important. The basis for the fairing process is a lines drawing usually in a scale 1:50 (1:48) or 1:100 (1:96). The curves of tangency of frames and waterlines to the flat bottom and to the parallel middle body must be specified. Drawings defining the shapes of stem and stern are necessary to determine the hull shapes in these regions.

About 400 points per ship, taken from these drawings, define the hull form to the computer. The fairing is performed automatically resulting in a table of offsets and the punched paper tape needed for automatic drawing of the body plan.

The lines thus generated by the computer may now be inspected by the shipbuilder whose possible changes are presented to the computer as new or altered point coordinates. The computer changes the lines accordingly. Through this repetitive procedure a body plan is produced, satisfying the requirements of the naval architect. The final body plan is stored in the computer for later reference.

The generation of the contours of the various parts is done by a computer programme package. The computer input is a coded description of the shape and position of each part. This code consists of English words and dimensions taken from the steel drawings. As will be shown later, the shape and position in the ship's hull of the various parts are easily defined by means of the autocode. In the computer, the autocode will serve as a basis for the calculation of the exact shape of the different parts. During this calculation the previously stored information on hull form, standard contour details, etc. will be used by the computer whenever this is necessary for the computation of the numerical description of the part contour. When requested, the computed part contour description is transformed into ESSI - parameters, and a corresponding control tape is produced. By means of this ESSI-tape either an accurate scale drawing or an automatically cut part is obtained. All the generated part contours are stored in the computer together with their identification numbers. In this way a complete library both for hull shape and all the individual parts is gradually built up in the computer.

A combined manual and automatic procedure for numerical nesting of parts on a steel plate has been developed, that should satisfy the requirements to efficient utilization of raw materials. This nesting procedure is also a part of Autokon, and the computer programmes needed for this purpose have been included in the programming system. As will be described in more detail later, a manual operation is needed to determine the position of the parts on the steel plate. A record is made of their relative position together with the part identification numbers and coded information on desired punch marking and cutting sequences. This serves then as the input to the computer and is used together with the part information already stored to produce the ESSI tape for the automatic flame cutting of the complete format.

The Autokon System as outlined above, will to a great extent simplify the work involved in the preparation of the input data needed for numerical flame cutting of ship plates. The system can be operated by members of the yard's staff with no special background neither in mathematics nor computer techniques. Instruction courses of a few weeks duration will enable these people to become efficient Autokon operators.

2. The Computer Operation

In order that an integrated computer system should be useful in shipbuilding, it must be applicable to the average shipyard of normal standard. This puts some heavy requirements on the system regarding flexibility, compatibility, ability to expand, simplicity in use, productivity and economy.

In the Autokon system these requirements are satisfied to a very great extent. Autokon is capable of handling a great variety of ship plates. Practically no changes are necessary in the organisation of the yard, its departments, or work routines as Autokon makes use of the same basic material such as drawings, lists, etc. There are no requirements regarding special qualifications of the personnel. Shipbuilders, e.g. loftsmen and naval architects, need only very little additional training outside of shipbuilding in order to operate the system. Practical experience has shown that this extia training is obtained by giving the operators an instruction course of two to three weeks duration.

The computer system developed to these specifications is fairly large. The present version of the Autokon programmes consists of approximately 100 000 computer instructions. The system is at present programmed for the Univac 1107 computer, but there is, of course, nothing to prevent the adaptation to other computers, provided they are of a sufficient size. Work is now being carried out to convert the Autokon programmes to the IBM 360 series computer. With exeption of Model 20 all other models of the IBM 360 series, from Model 30/64K and upwards, can be adapted for the Autokon System.

In its present version the system is divided into three programme packages:

a. Programme for hull definition.

b. Programme for part contour generation.

c. Plate nesting programme.

These programmes are linked together in such a way that full mutual utilization of results and of computing facilities is obtained. Each one of the three programme packages may, however, be used separately. The addition of new programme packages to the system is a fairly simple matter. Work is continously carried out in this direction by the development group at the Central Institute in Norway. The aim is to incorporate new programmes when they are sufficiently reliable and efficient for production use.

a. Programme for hull definition

A large portion of ship parts and details have contours which partly are determined by the shape of the hull. It is, therefore, essential to have a description of the hull form stored in the computer in order to facilitate computer determination of part shapes from part descriptions or part programmes which should be as simple as possible.

The purpose of the programme for numerical hull definition is, therefore, first of all to transform the conventional description of the ship's hull, i.e. a body plan or a lines drawing, into a numerical form suitable for further computer processing. It is also realized that it is equally important to reduce the amount of work needed in the conventional procedure of designing a satisfactory body plan. The numerical hull definition programme, therefore, performs both the fairing of the ship lines and the transformation of these lines into the required computer representation.

This programme differs to a certain extent from the lines fairing programmes that now are being used by the yards in different countries. Among other things, we believe that the mathematical method underlying the fairing process is a novel one. The method is based upon the manual method used by the loftsman for the fairing of ships' hulls. In other words, a mathematical model has been built in the computer that very closely simulates the fairing procedures that are used on the mouldloft.

In normal practice, waterlines, sections and buttocks are usually faired in separate steps, and more or less regarded as independent lines. In each of the fairing steps, however, the point coordinates from the preceeding fairing are used, and the results of the fairing in one projection are checked in the others. The procedure is repeated until the changes after each step are sufficiently small, and the naval architect is satisfied with the hull form.

Thin, elastic splines of different shapes are normally used for the fairing of the separate lines. The splines are forced through the given points, and as a rule kept in place by means of weights that are placed on top of the spline. The fairing of the line consists basically of removing one weight at the time in a sequence and allowing the spline to move slightly over a small portion of its length. The line is considered smooth when this procedure no longer has any effect on the spline curve.

By regarding this as a mechanical phenomenon it has been possible to design a mathematical model of the spline fairing. The model is built on the assumption that the relative position of the given points determines the crossection of the spline in that area. Furthermore, the elastic energy stored in the spline by virtue of its deformation shall be as small as possible. The latter assumption corresponds to the condition of the physical spline at the end of the manual fairing process. The computer fairing process which is based upon the mathematical model of the spline closely follows the manual procedure previously outlined. The fairing is performed separately for the forebody and the afterbody, each requiring the coordinates of about 200 points as input to the computer. The points define the hull form and also the boundary curve. This composed of the stem or to the stern curve, the curves of tangency to the flat bottom and to the parallel middlebody, and the highest waterline.

In the computer the process starts with the fairing of the boundary curve. When this has been finally determined the computer performs the independent fairing of a selected number of waterlines. From these lines offsets on a specified number of frames are calculated and used as basis for the frame fairing.

The faired frames again form the basis for the generation of new offsets and new fairing of waterlines. The procedure repeats until the computer has arrived at a set of lines which can not be improved by the computer with the given set of input data. The computer will then print out a table of offsets and simultaneously punch out an ESSI-tape for automatic drawing of the body plan. Finally, the computed description of the hull form is stored on magnetic tape in the computer for later reference.

b. Programme for part contour generation

The computer generation of the contours of plane parts is in principle an easy mathematical problem. However, the requirements on simplicity and economy of the system complicate the part generation programmes. The part contour generation programmes in the Autokon System make extensive use of crossreference to other information sources. Necessary computer routines have also been included for the interpretation of the Autokon programming language or autocode. These features are necessary in order to satisfy the requirements previously mentioned.

The autocode itself has been constructed for the purpose of being as simple and easily understandable as possible. Since this language is the communication link between the shipbuilder and the computer, it must be of sufficiently general nature to facilitate the description of all types of parts and processes the computer should handle. In order to make it legible to the shipbuilders, the autocode is based on a series of words from standard technical terminology. A description of the part contour is formed with these words together with dimensions and numbers taken from the engineering drawings.

The vocabulary of this language contains different types of words such as geometrical terms and words taken from the shipbuilder's terminology. Examples of the former are STRAIGHT LINE, CIRCLE, TANGENT, STARTPOINT, DIRECTION, LENGTH, ENDPOINT etc. By means of these terms and coordinates, geometrical curves can be quantitatively defined. A straight line between two points with coordinates x_1y_1 and x_2y_2 respectively is thus specified as:
STRAIGHT LINE: STARTPOINT (x_1y_1) ENDPOINT (x_2y_2).

The words from the shipbuilder's terminology have been included to simplify the description of ship parts. Words like FRAME, CUTOUT, MANHOLE, BRACKET, WEBFRAME etc. can be used directly. By means of these words complicated contour details and even complete parts are easily described to the computer.

The descriptions in autocode of the different parts are punched on paper tapes or cards and read into the computer which performs the necessary operations to give out the information required. Based upon the autocode the numerical descriptions of the part contours are generated and stored, and an ESSI-tape is produced for each part.

c. Programme for plate nesting

Many ship yards today have applied production methods which require that several parts be nested on one steel plate. The nesting procedure which is performed prior to the plate cutting, must be sufficiently efficient. This means that the manual effort should be as small as possible and the plate utilization as good as possible.

In order to meet the requirements from the yard, a special Autokon nesting procedure has been developed. As mentioned, in Autokon we combine manual and automatic operations. A manual determination and recording of the position of the parts on the steel plate has been found to be the most economical. However, the computer can be used at an advantage for the generation of the final ESSI-tape. This is used as input tape to the numerically controlled flame cutter for the automatic production of all the parts on the steel plate.

The information about the position of the parts, cutting and punch marking sequences etc. is read into the computer in the form of a table by means of punched cards or tape. Each part is referred to in the table by the same reference number under which the part previously has been stored by the part generation programmes.

3. Practical use of the Autokon System

It may be apparent from the foregoing that the Autokon System is applied to some of the more important activities necessary for the design and construction of ships' hulls. This is illustrated by fig. 12 which shows the activities from basic design to flamecutting. The activities in the design offices are shown as squares in the upper row of the diagram. They are meant to include the work going on in the various drawing offices for basic hull design, for steel design, and for making either optical scale drawings or N/C-tapes necessary for automatic flamecutting of ship details.

The figure illustrates how the computer is applied to these activities when the Autokon System has been adapted. Both the input/output and the computer operation and storing of results are shown underneath and the various lines between blocks illustrate the functional connection between the design activities and the computer operations.

The system is operated and controlled by the shipbuilders, who on the whole perform the same type of work as they do in the conventional ship yard. The difference is mainly that the laborious and time comsuming work is left to the computer, and the manual work to a much higher degree consists of job specification, control of operations and checking of results. A short description of the system at work will illustrate this.

The use of Autokon for the plate manufacture starts with the lines fairing. From an ordinary lines drawing in a scale (quarter scale, or 1:50), a coarse table of offsets is recorded by the loftsmen. This table containing about 200 points for one half body defines completely the hull form to the computer including the stem or stern, shape of upper deck, "knuckle curves" in the hull surface etc.

When this table is fed into the computer, the automatic lines fairing is performed, giving out an ESSI-tape and a complete table of offsets for all desired water lines, buttocks and frame stations. From the ESSI-tape a body plan is automatically drawn by means of a numerically controlled drawing machine (fig. 2) The body plan and the offset table are now inspected by the shipbuilder, who will advise possible corrections if the lines are not satisfactorily smooth or have other deficiencies. The corrections are then fed into the computer, and the whole sequence is repeated, possibly ending with a new set of corrections to the last body plan. This procedure continues until the naval architect is satisfied with the lines generated by the computer and the drawing machine.

The computer fairing of body plan represents a great saving in time and manual effort. Depending on the quality and consistency of the preliminary lines drawing, a period from one to two weeks is needed for a complete ship, and the quality of the body plan is at least as good as with manual fairing (fig. 3). The fairing covers the complete hull including the stem and stern regions, so that most of the manual work has been eliminated.

The numerical description of the body plan which thus has been stored in the computer is one of the prerequisites for an efficient use of the part generation programmes. In addition, other types of necessary information will be specified by the shipbuilder and read into the computer. As previously mentioned these may be either coordinate tables defining the position of certain structures inside the hull, or descriptions of standard contour details such as brackets, holes etc. The specification of such information is normally made as early as possible during the design phase. In any event, the data should be in the computer before the part coding starts.

The writing of the autocode for the various internal ship parts (fig. 4) is as a rule performed by people from the mould loft or the optical drawing office. The autocode is specified from the usual steel drawings now employed by the yards. The writing of the autocode is in general an easy and straight forward procedure. Basicly three steps are required:

i) Define the hull's yard number, the part's identification number, and the position in the hull.

ii) Subdivide the contour into elements that can be defined by means of the Autokon language.

iii) Write down the autocode for the different elements in the correct sequence.

By means of this procedure the autocode may be written down for all internal ship's parts with comparatively little effort. In general this method has been found to compare favourably with other methods of template making. An example of an autocode for a floor part is given in fig. 5, and it may be apparent that the writing of the autocode is not a big job. It should also be emphasized that the autocode easily lends itself to further rationalization. This makes possible a method of part specification which contains only what is absolutely necessary for the definition of each plate. This so called parametric autocode is used for ship parts for which there are many of one type. For instance, most of the brackets, web parts and floor parts are specified by means of parametric codes. For such parts, which constitute the majority of all internal parts, the coding is now extremely simple and very little time is needed for their specification. Fig. 6 shows the parametric code for a floor part, from which it is seen that less than 10 numbers are needed for the complete specification of this part. Similarly fig. 7 shows examples of the parametric codes for brackets.

As stated earlier, the computer will generate the exact shapes of the different parts from these autocodes, store the corresponding part descriptions under their reference numbers and punch out ESSI-tapes for automatic drawing and flamecutting. Practical experience has shown that these scale drawings of the parts are rather essential, because this gives the shipbuilder an efficient and easy way of checking the ESSI-tapes before a possible production of the plates.

The scale drawings also serve another purpose in all cases where part nesting is employed. Here the drawings are a necessary basis for the whole procedure which briefly consists of the following steps:

i) The computer produced ESSI-tapes are drawn to scale by means of the drawing machine.

ii) A manual operator solves the puzzle of fitting these drawings together on a scaled representation of the steel plate so as to obtain minimum scrap.

iii) By means of the standard Autokon recording procedure the position and orientation of each part, the cutting and punch marking sequences are recorded together with the identification number of the parts.

iv) The recorded information is used by the computer to produce the ESSI-tape for the nested parts.

v) After checking by automatic drawing, the tape is used for the numerical flame cutting of the required plates.

The procedure as outlined can easily be performed by the yards' own staff. It will be noted that the manual activity involved in this procedure is limited mainly to the determination and recording of the position of the various parts together with the checking of final results. In order to make these activities as efficient as possible a special nesting table has been deviced. The table shown in fig. 8 is fitted with digital readout and also automatic printing of the readings in order to speed up the recording of the part positions. The nesting procedure, as illustrated in fig. 9, is generally found to be an easy operation. With this equipment the recording is also greatly simplified, and for the average steel plate the manual nesting and recording is performed in a matter of minutes. In addition to this, the method is found to be very flexible. Different alternatives for part positions on the steel plate are quickly made, and a good plate utilization is, therefore, easily obtained. (fig. 10).

The tapes produced in this way are now complete, and they are sent to the plate shop. Here they are used for the automatic production of all plate parts in numerically controlled flame cutters. As mentioned earlier the ESSI N/C-system is used for the control of the flame cutting machine, as this system is especially adapted to this type of work.

A machine configuration as illustrated on fig. 11 is often found to be the most advantageous, especially in the beginning where both optical and numerical flame cutting is used by the yard. It is seen that to change from optical to numerical flame cutting merely involves the throwing of an electric switch, and thereby gives the operators full freedom of choice regarding the mode of operation of the flame cutter.

In the numerical work the flame cutter operation is very simple indeed. The operator's job is only to place the tape in the numerical director, place the cutting torch in the correct start position, preset the correct cutting speed, and press the start button. From then all operations are performed automatically including flame and speed control, punch marking etc., and the operator has only to check that everything is functioning correctly.

This is one of the reasons why the numerical control of flame cutting has been so well received by the shipyards. But in addition to the ease of operation, additional advantages have been experienced of which higher productivity and greater cutting accuracy are the most predominant. An increase in productivity of the flame cutter between 5% and 25% has been registered and due to the higher accuracy of the finished parts, savings in assembly are obtained.

4. Status and Future Possibilities

Numerical flame cutting and computer aided design and preparation have definitely provedthoir value. Our own experience with Autokon through several years of practical use has revealed several advantages, the most important of which may be summarized as follows:

1. Increased accuracy of the finished parts.

2. Increased productivity of flame cutting and assembly of ship details.

3. Increased efficiency of steel design and work preparation especially on the optical/numerical drawing offices. Here a reduction of qualified personnel may easily be obtained.

4. Reduction of time needed for the various design activities. This is especially so with hull fairing, where a complete body plan can be finished within one to two weeks.

5. Material savings both from better utilization and also because reliable design results may be available early enough to form the basis for the steel orders.

6. Good possibilities for further expansion by including more activities in the Autokon system.

Regarding the last point there are a great many new activities that can be incorporated in the system. Of the most obvious ones may be mentioned the inclusion of shell expansion process by computers. A shell expansion programme has been written and is now being tested in production. It is believed that this programme will be incorporated in the system during the autumn of the year 1968.

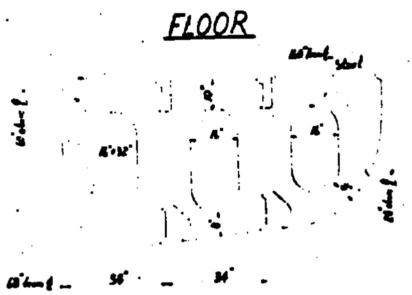
Another possibility is the utilization of modified hull fairing programmes for the basic design of the ship's hull. Our experiments have shown that the programmes may be used at an advantage at least for the design of hull forms from one of the standard hull series, such as the Todd 60 series. Here the hull fairing programmes coupled with programmes for design calculations can do most of the work which now is carried out by the naval architects. The computer can simultaneously deliver a lines drawing with the accompanying hydrostatic data, and the faired body plan. It is thus possible to finalize the hull form at a very early stage of the design.

The most interesting new application of Autokon seems to be in the steel design offices. In fig. 12 was illustrated that Autokon now mainly is used in connection with the activities in the optical drawing offices. It has been demonstrated, however, that only slight modifications in our part contour generation programmes will make them well suited also for steel design work. As shown in this diagram, the procedures for steel design will be very similar to those now used in the optical/numerical drawing office. The difference will mainly be that the computer will do the design work of larger parts of the ship, such as complete web frames, the double bottom with all its parts etc. It is a straight forward procedure to let the computer and drawing machine produce drawings of such parts. In this procedure it is also possible to include stress calculations, determination of scantlings, design optimization etc., for which many computer programmes already exist. This will then reduce the amount of manual drawing needed and greatly speed up the design phase.

What is equally important is that one with these extensions has obtained a uniform design method, from the basic design to the production of the ship details. It should also be stressed that one now has built up in the computer a complete library, containing all information of the ship and its details. Furthermore, this information is generated at the earliest possible stage. In addition to the savings in time and manpower thus gained, one has also the advantage that a detailed knowledge of the ship is available sufficiently early to be useful for other purposes.

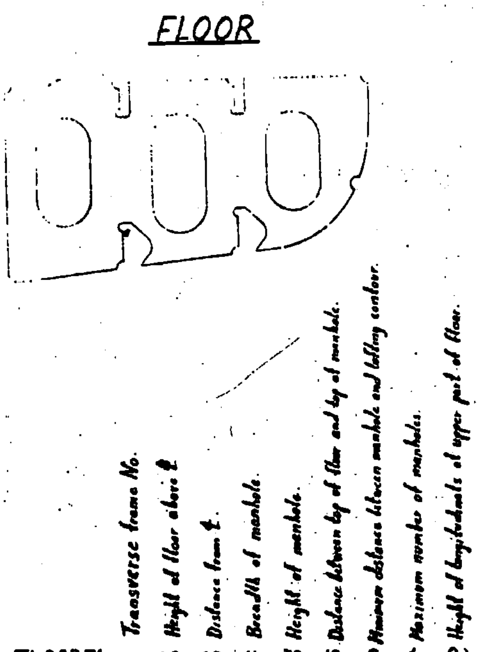
Production planning is perhaps the field which most obviously may profit from this new source of information. Computer planning scheemes are now being developed. We have clearly seen that a computer linkage of the design and planning activities will result in additional advantages that hardly can be obtained in any other way.

A computer system including all of the activities mentioned may still require a few years of development before it can be realized. Work is, however, continuously going on along these lines, and there is no doubt that these systems will be realized in a not too distant future.



```
X  YN536 IDN277 TFR 40>    |CUTO18(•136•60•180•9•2)•
GENPAR(•16) • SCON•        |SL:PT(•68•60) DIR(•0)•
RAP SPT(•160•60)DIR(•0)L(•0)>|INT(•200•60)•IT(ON•INT(•68•60)•
HOLE11(•153•50•0•16•8•36)• |SL:PT(•68•0) DIR(•90)•
HOLE11(•119•50•0•16•8•36)• |CUTO1(•68•60-135•2)•
HOLE6(•85•50•0•16•32)•     |SL:PT(•68•60)DIR(•0)L(•0)•
RAP:EPT(•68•60)•           |RAP:EPT(•68•60)•
SL:PT(•68•60) DIR(•0)•     |ECON •
CUTO18(•102•60•180•9•2)•   |ESSI(IDN 277 FLOOR)•
SL:PT(•68•60) DIR(•0)•     |FINISH • X
```

Fig. 5. Floor plate with corresponding autocode.


FLOOR
Transverse frame No.
Height of floor above ℄
Distance from ℄.
Breadth of manhole.
Height of manhole.
Distance between top of floor and top of manhole.
Minimum distance between manhole and bottom contour.
Maximum number of manholes.
Height of longitudinals at upper part of floor.

```
FLOOR7(•40 •60 •68 •16 •32 •10 •8 •6 •9)
```

Fig. 6. Floor plate with the corresponding parametric autocode.

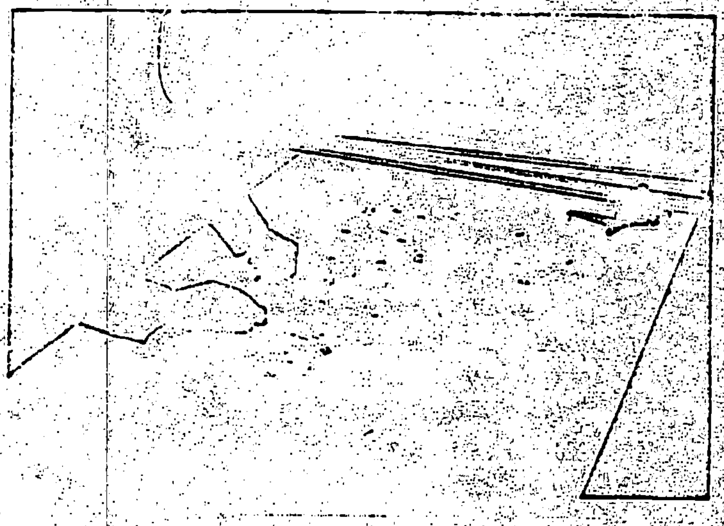


Fig. 9. Nesting procedure.

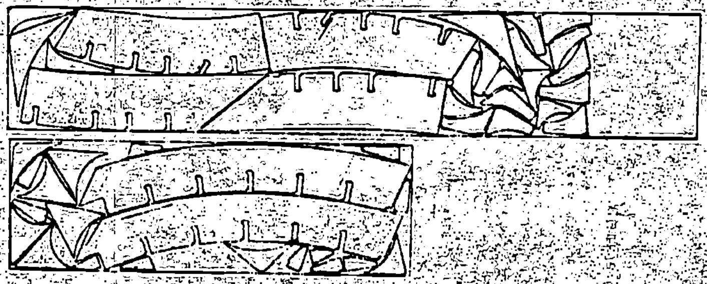


Fig. 10. Examples of nested plates.

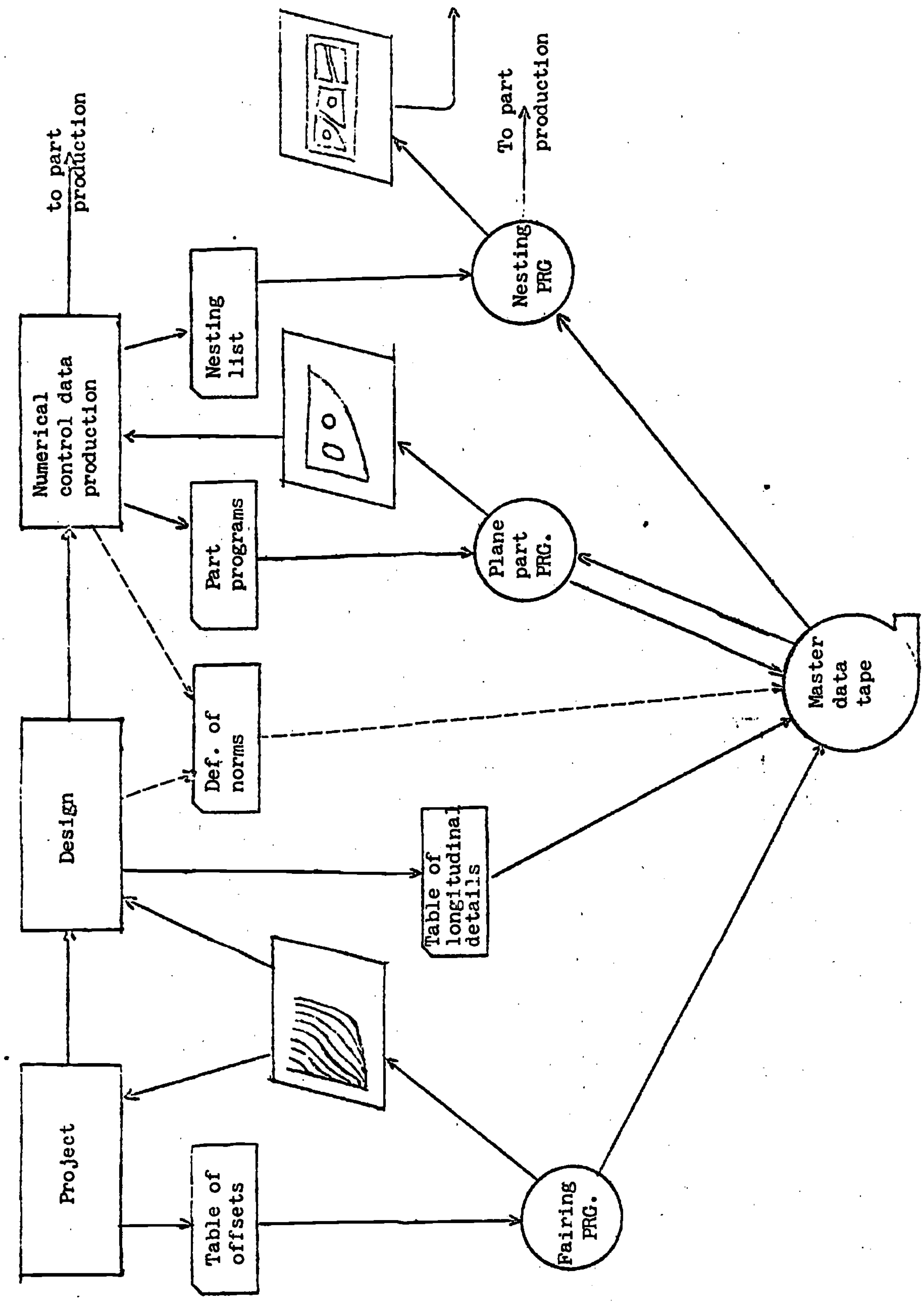

Project
Design
Numerical control data production
to part production
Table of offsets
Def. of norms
Part programs
Nesting list
Table of longitudinal details
Fairing PRG.
Plane part PRG.
Nesting PRG
To part production
Master data tape

Fig. 12

4.5 References 95 through 106.

95. Evans, H., Khoushy, D., "Optimized Design of Midship Section Structure," SNAME, 1963.

96. Evans, H., "Optimization of Midshipsection Structure," Computer-Aided Ship Design and Construction Symposium, Washington, D.C., August 1966.

97. Chapman, K. R., "Some Aspects of the Application of Electronic Computers to Ship Design: Midship Section Scantlings," Transactions RINA, 1967.

98. Buxton, I. L., "The Design of Tanker Hull Structure by Computer with Particular Reference to One Midship Cargo Tank," Transactions RINA, 1966.

99. Nachtsheim, J. J., Romberg, B. W., O'Brien, J. B., "Computer Aided Structural Detailing of Ships (CASDOS)," SNAME 1967.

100. Romberg, B. W., "CASDOS—A System for the Computer Aided Structural Detailing of Ships," contribution contained Ref. 36.

101. Cohen, J. B., Gardner, G. O., Romberg, B. W., "Design Automation in Ship Detailing," Proceedings A.C.M. National Meeting, 1967, reprinted in Ref. 36.

102. Romberg, B. W., "A Computer System for Structural Detailing of Naval Ships," Chesapeake & Hampton Roads SNAME section paper, September 1968.

103. Hysing, T., "From Basic Design to Flamecutting," internal note of CIIR, Oslo, January 1968.

104. Ulrichsen, B. B., "Application of Numerical Control to Flame Cutting Machines," Colloquium "Thermal Cutting" of International Institute of Welding, Annual Assembly, Paris, 1965.

105. Sorensen, P. F., "Autokon and Its Further Possibilities," Tanker and Bulk Carrier, May 1968.

106. Anonymous, "How a Progressive Yard Meets Workload," Marine Engineering/Log, August 1967.

5. Computer Graphics and Numerical Control.

5.1 Introduction.

Any review of computer-aided design would be incomplete without discussing the eminent role of computer graphics and numerical control. Engineers are used to present their design ideas in drawings and they communicate with the production department by means of working plans so that the significance of graphics in design need not be further proven. But as for numerical control this is less obvious since it serves the production rather than the design.

It would nevertheless be the gravest mistake should the designer ignore this important aspect of production leading to a natural completion of the process he started. On the contrary, computer-aided design can only live up to its full potential where it is followed by computer-assisted production, and vice versa. It would be inconsistent, for example, using a digital drafting machine in the design office, going to the trouble of preparing complex and voluminous input data, but then stopping short of using the output of the same process in automating production.

We must view the chain of events from the engineer's idea to fabrication as one continuous process, Fig. 29. The computer can assist in many ways: In sketching up the first thoughts if active, lightpen-controlled graphics facilities are available; in evaluating and optimizing the various, gradually improving versions of the design; and finally, in preparing working plans and numerical control tapes for fabrication. The striking fact is that through all phases of this sequence we need essentially the same information, preferably in immediately compatible form: A definition of the shape of the object and how it is made.

This may explain why computer graphics and numerical control have so much in common. Both are based on the medium of digitized information on geometry. Both use very similar technology. Both have been thriving best in environments where the computer systems approach is at home, notably in the U.S. aircraft industry.

Historically, developments in numerical control of machine tools began soon after the appearance of digital computers (J. T.

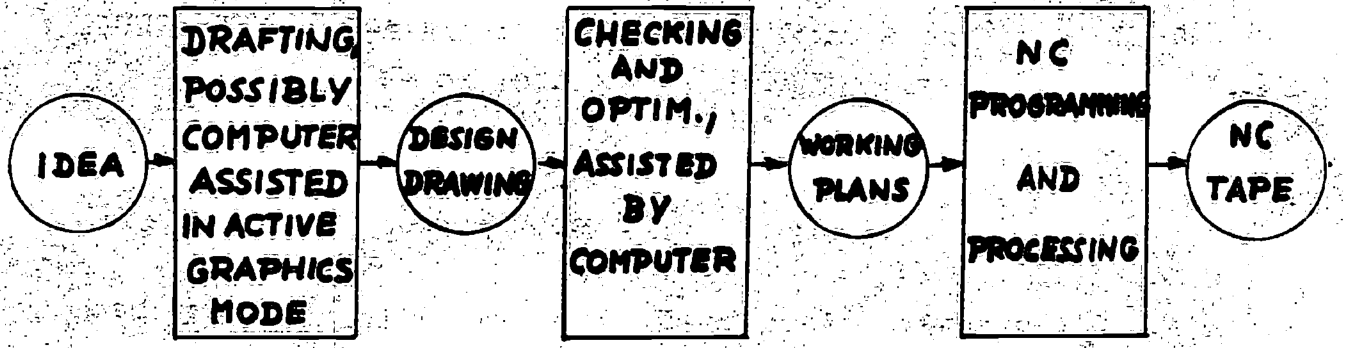

IDEA
DRAFTING, POSSIBLY COMPUTER ASSISTED IN ACTIVE GRAPHICS MODE
DESIGN DRAWING
CHECKING AND OPTIM., ASSISTED BY COMPUTER
WORKING PLANS
NC PROGRAMMING AND PROCESSING
NC TAPE

Fig. 29: Computer-aided design process.

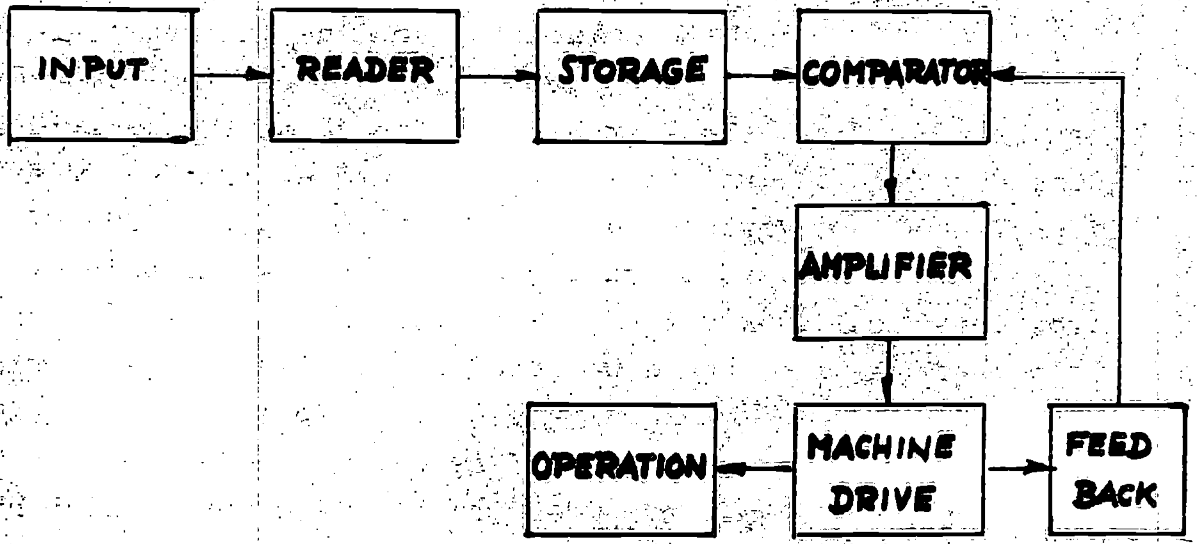


Fig. 30: NC feedback system.

Parsons in collaboration with MIT, ca. 1948). This resulted in what is believed to be the first successful application of numerical control, a three-motion contour-path milling machine (1952). Another major step forward was that through very substantial support of the U.S. Air Force the necessary versatile and comprehensive numerical control programming languages became available in the later fifties, notably APT (Automatically Programmed Tool, originated at MIT, and further expanded by the Aerospace Industries Association and IIT Research Institute). In the commercial field, NC machines have been gaining rapidly in the market since about 1960.

In computer graphics some pioneering work also dates back to the fifties, but it was not before the last five years that wider segments of the industry began to be attracted to plotters of gradually improving accuracy and to interactive computer graphics.

In shipbuilding, flamecutting has been successfully automated by NC, for example, in the systems of British Oxygen Co. and Ferranti Ltd., AUTOKON and CASDOS, Refs. 99, 104, 111, 112. These systems also incorporate graphic display capabilities.

5.2 Principles of digital plotting and numerical control.

5.21 Definitions.

Numerical control of a system is the control by means of alphanumeric data given as input via a suitable medium. The system is usually a feedback system whose path and rate are to be controlled. The input medium is generally paper tape or magnetic tape. Fig. 30 shows the components of the system.

There are two basic types of NC:

*Point to point control (discrete positioning)

*Continuous path control (contouring)

Point to point control (PTP) requires only that the device be moved from a given point to another given point without prescribing the path. (The path has an effect on time economy, but accuracy is irrelevant in these applications.) Only discrete point data are needed as input. Such operation would be appropriate for a drill press.

Continuous path control (CP) is much more demanding. There the device must be guided continuously along a given curve within a specified tolerance.

Certain PTP machines allow the compromise of a straight cut motion, i.e., a straight guided motion along one coordinate axis at a time.

Coordinate information is given to the system either in absolute or incremental form. Point to point control is usually based on absolute coordinates. Continuous path machines generally take incremental commands, that is, each movement makes reference to the outcome of the preceding one.

NC equipment is classified by the number of axes that can be controlled in a coordinated fashion: 2 to 5, in exceptions more. Note that table and/or tool (stylus) may be the moving part.

The basic capabilities of the contour systems are limited to the elementary functions:

*Linear interpolation

*Circular interpolation

*Parabolic interpolation

Curved portions of the contour must be approximated in one of these three manners. Linear interpolation is often the only available feature, parabolic interpolation is rare.

Finally, as an example, we may classify digital plotters as 2-axis continuous path systems which is also true for most flame cutting tools (without bevel feature).

5.22 The programming process.

In numerical control the instructions for the tool (or stylus) operation have to be encoded on tape in terms of digital symbols the machine can respond to. This process is called part programming as opposed to general purpose computer programming.

Part programming may be done manually, at least in simple cases, or with the aid of a computer translating a symbolic code. In both cases a definition of the part has to be given first. This is done in terms of point coordinates, dimensions, and mathematical shapes where necessary. Dimensioning is organized around functional tolerances (point distances where they matter for the performance) rather than in the usual base line dimensioning style.

A complete plan for the fabrication of the part has to exist also. This includes information as to when the tool is operating, coolant on, and when it is in the rapid traverse mode etc. At this stage the programmer will prepare a complete step by step plan of the operation, the manuscript of the program. The manuscript contains, in some pre-determined sequence and form, information as to starting and unloading point of each move (direction), starting and stopping the machine, feed rates, and tool speeds, and the various auxiliary functions (coolant, etc.).

Where complex curves are involved the program must generate an approximation in terms of elementary steps of the machine. This is usually done with the aid of computers (postprocessors).

5.23 Computer programs for NC programming.

NC compilers are computer programs that translate the "English" type instructions of the part programmer (symbolic code) into coordinate and other control instructions for the machine. However, due to differences in machine hardware and dynamics, special postprocessing (by a separate computer program) has to take place thereafter.

Some of the major symbolic NC programming languages are:

*AUTOPROPS: PTP, 2-axis, IBM, Pratt & Whitney
*AUTOSPOT: PTP, 3(4)-axis, IBM
*CAMP I: PTP, 3-axis, Westinghouse
*CAMP II: PTP, 2 to 5-axis, Westinghouse, IBM
*PRONTO: PTP, 3-axis, GE
*SNAP: PTP, 2-axis, IBM
*APT, APT III: CP, multi-axis, MIT, IIT, and others
*ADAPT: CP, 3-axis, IBM
*AUTOMAP: CP, 2-axis, IBM
*AUTOPROMT: CP, multi-axis, IBM
*SPLIT: CP, multi-axis, Sundstrand, IBM
*SYMPAC: CP, 3-axis, MIT, Univac
*AUTOKON (autocode): CP, 2-axis, ESSI, Kongsberg

For contouring applications APT III has assumed by far the greatest significance. APT part programs are written in a vocabulary consisting of specifications as to:

*Tool shape

*Tolerances

*Direction of motion

*Tool position relative to controlling surfaces

*Geometric definitions (Points, lines, circles, conic sections, cylinders,...)

*Auxiliary commands (feedrate, speed,...)

*Computation (addition,...,sine,...)

*Macros (canned instruction sequences)

*Loops

The output from APT III (and its postprocessors) is a sequence of digital instructions in Binary Coded Decimal code, usually stored on 8 level tape following EIA and AIA standards. In this medium the elements of information are words of several characters. The meaning of the words may cover:

*Sequence number (ID)

*Preparatory function (acceleration)

*Feed rate (FRN)

*Speed (of drilling tool, etc.)

*Tool (tool ID)

*Miscellaneous function (coolant, etc.)

Each word type is identified by a lead character (x, y, z,...): "Word address format."

5.24 NC drafting machines.

The following lists a few examples of computer graphics devices and some of their characteristics. For more details, see Ref. 110.

*Orthomat:

Principle: Digital

Curve capability: Second degree curves

Size: 5 by 12 feet

Drawing speed: Up to 200 ipm

Tolerance: ±0.005 in.

Universal Drafting Machine Corp.

*GE Autoprogrammer:

Principle: Digital plotter in reverse. Reads out stylus positions digitally.

Size: Up to 5 1/2 by 15 1/2 feet

Accuracy: ±0.0005 in.

Output: Standard tape

Scale: Optional

General Electric

*Gerber Series 600:

Principle: Digital plotter

Size: Up to 4 by 5 feet

Speed: 200 ipm

Tolerance: ±0.005 in.

Gerber Scientific Instrument

*CalComp:

Principle: Digital plotter, using paper drum

Curve capability: Lines

Paper width: Up to 2.5 feet.

Speed: About 180 ipm

California Computer Products

*Kingmatic:

Principle: Digital plotting table

Curve capability: Lines, circles, parabolas

Size: Up to 5 by 8 feet

Speed: 200 ipm

Tolerance: ±0.0005 in. (for smaller models)

Baldwin Kongsberg Company

*Cathode ray tubes:

Purpose: Direct viewing, active graphics, photographs to get hard copies

Input: Programmed data

Active control: By light-pen

Examples: S-C 1090 direct viewing device,
S-C 4020 computer recorder
(General Dynamics Corp.)

5.25 Active graphics systems.

A variety of design-oriented, light-pen controlled, active graphics systems has been developed during the last decade: MIT's Sketchpad (Sutherland), General Motor's DAC-I, Lockheed Georgia's Man Computer Graphics, etc. Ref. 36 gives an introductory review and many other references.

5.26 Advantages of numerical control.

General:

1. Productivity: Significant reduction of idle time.
2. Accuracy and repeatability: Excellent (±0.001 in.)
3. Tooling: Savings in jigs, templates, etc.
4. Scrap: Minimized, no human errors
5. Quality control: Less critical.
6. Capital: Less required since fewer machines, less floor space, etc.
7. Inventory: Smaller since setting-up faster
8. Lead time: Reduced.
9. Versatility: Better than human for complex jobs
10. Scheduling: More reliable.

Comparison with analog control:

NC is:

More accurate
Faster in operation
More versatile

5.27 Economics of NC.

Numerical control equipment is generally more expensive than conventional machines, not seldom by about 100 percent. But this can frequently be offset by the savings already indicated. However, as lot size gets large (mass production) special purpose machine tools tend to be economically superior to NC equipment. In summary, NC has a chance to be competitive (Ref. 108) where:

*Lot size is moderate (say 100 pieces)
*Parts are complex
*Design changes are frequent
*Lead time is at a premium
*Tooling costs are high

*Inspection costs are high
*Production is decentralized (subcontracting)
*Scrapping is costly
*Floor space is critical

5.28 Future.

Both computer graphics and numerical control appear to be only at the beginning. More and more applications are becoming technically and economically feasible (welding, tube bending, controlling quality by inspection machines, testing electronic systems,...).

It has been estimated that before ten years more than half of all metal-cutting tools in the U.S. will be NC automated.

The shipbuilding industry, like most others, faces the problem of integrating all capabilities in automated design and production now becoming available into a single, most economical system.

Voices are now being heard that new developments in computer technology will soon render off-line numerical control obsolete in favor of complete on-line computer process control.

5.3 The use of a plotter and graphics console.

The discussion of these subjects will be based on material contained in the Calcomp Digital Plotting System Manual, Ref. 113, and descriptions of the DEC 338 system graphics subroutines DF and DRAWL issued by the Concomp Project, The University of Michigan.

5.4 References 107 through 117.

107. Dyke, R. M., "Numerical Control," Prentice-Hall, Englewood Cliffs, New Jersey, 1967.

108. Childs, J. J., "Principles of Numerical Control," Industrial Press, Inc., New York, 1965.

109. "Advanced Numerical Control," Book I, sponsored by the American Society of Tool and Manufacturing Engineers, Detroit, 1964.

110. Thornhill, R. B., "Engineering Graphics and Numerical Control," McGraw-Hill Book Company, New York, 1967.

111. McIver, T., "Computer-Controlled Flame Cutting," Shipping World, January, 1963.

112. Pearse, W. H., Sillifant, R. R., "Automatic Flame Profiling of Plate in Shipyard Practice," Welding and Metal Fabrication, December 1957.

113. "The University of Michigan Digital Plotting System," manual prepared by the University of Michigan Computing Center, March 1967, with numerous later revisions and supplements.

114. Herzog, B., "Computer Graphics: An Introduction," lecture notes, Engineering Summer Conferences, The University of Michigan, contained in Ref. 36.

115. Devere, G. S., Hargreaves, B., Walker, D. M., "The DAC-I System," Datamation, June 1966, reprinted in Ref. 36.

116. Chasen, S. H., "Experience in the Application of Interactive Computer Graphics," Lockheed-Georgia Company, reprinted in Ref. 36.

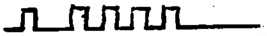
117. Reinhard, P., et al., "Application of Computers to Automated Design," lecture notes, Engineering Summer Conference, The University of Michigan, 1968.

6. Computer-Aided Design Languages and Systems.

6.1 Introduction.

Computer-aided design languages are problem-oriented computer languages and thus belong to the highest level languages, see Table V. They are extremely user-oriented.

Table V: Language Levels

Type of language	Example	Input	Output	Translator
Machine	IBM 360 code	bits = 011 110...		hardware
Assembly	FAP, UMAP	Op A, B, A, i, n	bits = 011 110...	assembler
Procedure-oriented	FORTRAN, ALGOL	A = A + B	Op A, B, A, i, n	compiler
Problem-oriented	ICES, CASDOS	FRAME (A, B,...)	A = A + B	processor

The purpose of such languages is to facilitate computer use by users who are not programming specialists for problems that cannot easily be pre-programmed. Engineering design is a typical example. While the basic steps that may occur can usually be anticipated, their sequence and interaction might not. Therefore, we want to give the designer control over the program at execution time.

This objective motivates the following properties of computer-aided design systems:

*Problem-oriented language: The language must be easily understood and handled by the user. It should be written in the terminology of his problem.

*Time-sharing: To have control at execution time it is practically necessary to give the user his own time-sharing terminal where he can interact with the system without interfering with other users. There are exceptions to this rule (CASDOS).

*Conversational mode: To exert meaningful control the user must have the opportunity to exchange information with the system in a question and answer mode.

*Display: Graphical output (and input) are often essential to speed up communication.

In many graphics-oriented problems, the system would be unattractive if the user had the burden of translating numbers into pictures.

The above functional objectives lead to certain technical requirements for the computer system:

*Flexible data management: Since the amount and type of data can vary unpredictably the system must have the capability of efficient data management (Convenient storage, fast retrieval, easy updating, convenient use of secondary storage, good user control of data structure).

*Flexible program management: In order to be able to access and execute any sequence of programs, subprograms, and subsubprograms, the programs must be highly modularized. The building blocks must be put together and provided with data automatically at execution time. This may be done under program control, or user control, or both.

6.2 Some computer-aided design systems.

6.21 ICES.

ICES (Integrated Civil Engineering System) is a pioneering effort and a model case in computer-aided design systems. It has been developed under the leadership of Daniel Roos, MIT Civil Engineering Systems Laboratory. It has the objective of creating a complete computerized civil engineering design environment. The following are a few highlights. For details, see Ref. 118.

System environment:	Time sharing, active graphics
Programming language:	ICETRAN = FORTRAN + command language
Command:	Format-free technical terminology
Program modules:	COGO (Coordinate Geometry), STRESS (Structural Engineering System Solver), and many others.
Data structure:	Arrays, lists, array-lists.
Data management:	Dynamic memory allocation, list processing, subsystem data transfer, automatic memory reorganization (update).

Functions of ICES executive: Analysis of commands, incremental execution of command program.

Command definition language: Permits defining new commands.

6.22 COMRADE.

COMRADE (Computer-Aided Design Environment) is a computer-aided ship design software system being developed by the U.S. Navy (NSRDC). A good overview is given by Maiorana in Ref. 119. Some of the main features are:

System environment: Time-sharing, active graphics

Language: FORTRAN + command language

Components of COMRADE: Data management system, program management system, executive.

Data management: Generalized from CASDOS AUTOFILE, similar to ICES. Dynamic storage allocation, list processing for complex structures, use of random access storage in hierarchical order.

Other capabilities: Command definition language,User-to-user communication.

6.23 CASDOS-AUTOFILE.

The CASDOS software and its executive system, AUTOFILE, are organized much like the systems mentioned above, in fact, CASDOS served as one of the models for COMRADE. A difference exists in that CASDOS is designed to operate in the batch mode, and AUTOFILE does not use random access memory. But it could be readily adapted. For a more detailed description of the system, see Ref. 101.

6.3 System organization.

The greatly increased functional capabilities users demand from modern computer systems necessitate far greater efforts in system organization. Table VI shows some of the principal differences between classic and modern systems.

Feature	Classic System	Modern System
Language	Procedure-oriented (FORTRAN)	Problem-oriented
Operating mode	Batch	Time-sharing
Secondary storage	Tape-oriented	Disk-oriented
Access to sec. storage	Sequential	Random
Data structures	Array	List
Storage allocation	Rigid	Dynamic
Updating	Difficult	Easy
User control	Before execution	During execution

Table VI: Comparison of Classic and Modern Computer Systems

In the classic FORTRAN-type system two major shortcomings are the rigidity of storage allocation and the lack of convenient data structure. Memory space is allocated before program execution, namely during compilation. If the program is to run with data sets of varying size, maximum size is allocated and much space is wasted on the average run. This drawback is eliminated in modern systems by dynamic storage allocation. Storage is allocated during execution and varies during the run exactly according to need. Besides, arrays are automatically shifted between primary and secondary storage so that the user may regard primary and secondary storage as identical without sacrifice in cost or efficiency.

In a classic system with tape-oriented secondary storage devices data structures are sequential. This means a string of data values (records) is stored in or retrieved from contiguous storage locations, and the physical sequence of the records furnishes the only information about their logical relation. Each record is identified by one attribute or key, and the records are stored in a file according to the value of the key. In retrieving the record from such a file the whole sequence of records must be checked until the key matches the search key. The data are structured according to only a single key. If the same data are of relevance in connection with a different attribute or key one has to sort the entire file and rearrange the records so as to make them accessible under the new search key.

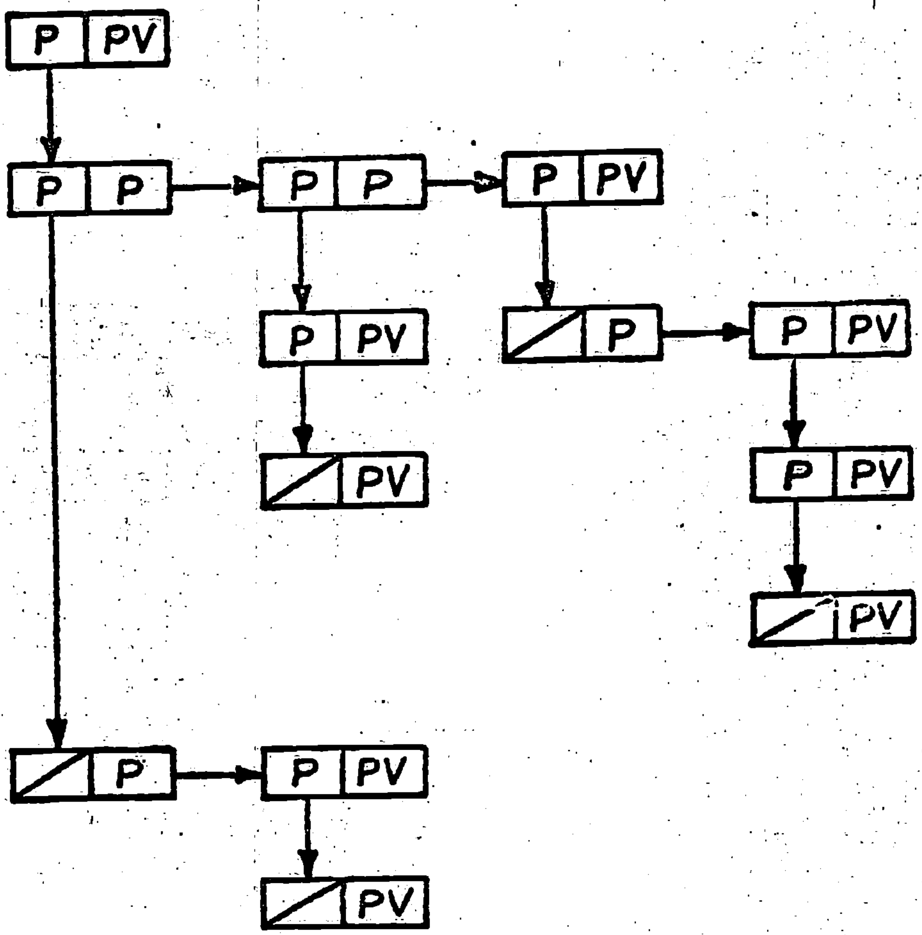
Modern data structures are oriented toward random-access secondary storage devices (disk). This provides essential, new logical capabilities, see Table VII.

Method	Single key access	Random access	Multiple key access	Cross-Associations
Sequential	Yes	No	No	No
Direct	Yes	Yes	No	No
Simple list	Yes	Yes	Yes	No
Ring structure	Yes	Yes	Yes	Yes

Table VII: Capabilities of data structuring methods

The direct method uses the random access device in the crudest manner. Every record has one key by which it can be searched just like in the sequential method. But the physical sequence of records no longer necessarily corresponds to their logical sequence, which permits introducing dynamic storage allocation. On the other hand one must have a system of converting keys into storage locations, for example, a table. This makes the handling of larger files rather cumbersome.

List structures use a system of pointers to divorce the logical and physical sequence of records. A pointer is a memory cell containing an address pointing to the next memory cell in the logical sequence. A list consists of any number of cell combinations connected by pointers. Fig. 31 shows a simple one-way list. In the example cell pairs are connected in such a way that records having keys of the same type are chained together in logical order. In a generalized form records may contain any number of keys each provided with a pointer so that the same data may be accessed using many different keys. All it takes is an index of entry points into the list for every key. This way of accessing the data by multiple keys makes file manipulation (updating, merging, etc.) very convenient.



LEGEND: P DENOTES A POINTER (ADDRESS)

PV DENOTES A PROPERTY VALUE

⧄ DENOTES TERMINATION MARK

FIGURE 31

LIST STRUCTURE SCHEMATIC

(SIMPLE ONE-WAY LIST)

(Figure reproduced from Kreitner, Ref. 120)

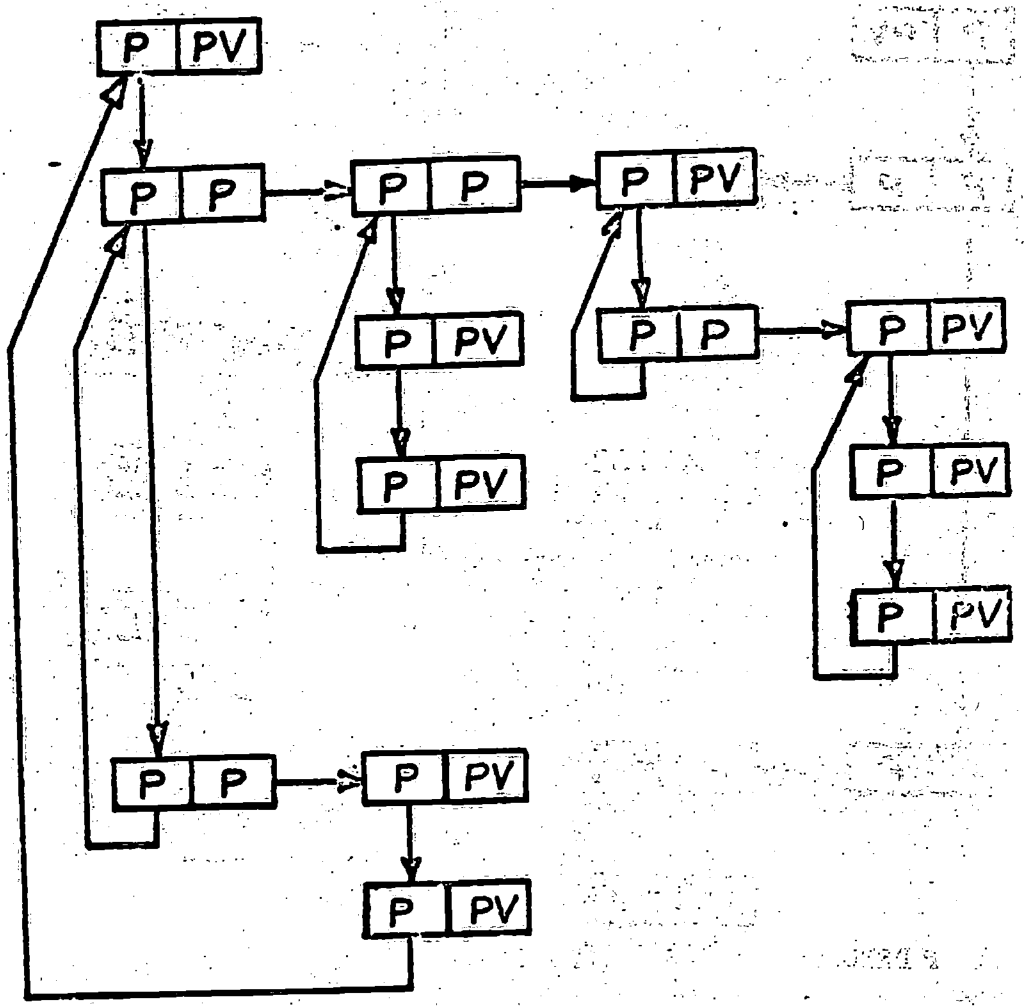


FIGURE 32

RING STRUCTURE SCHEMATIC

(Figure reproduced from Kreitner, Ref. 120)

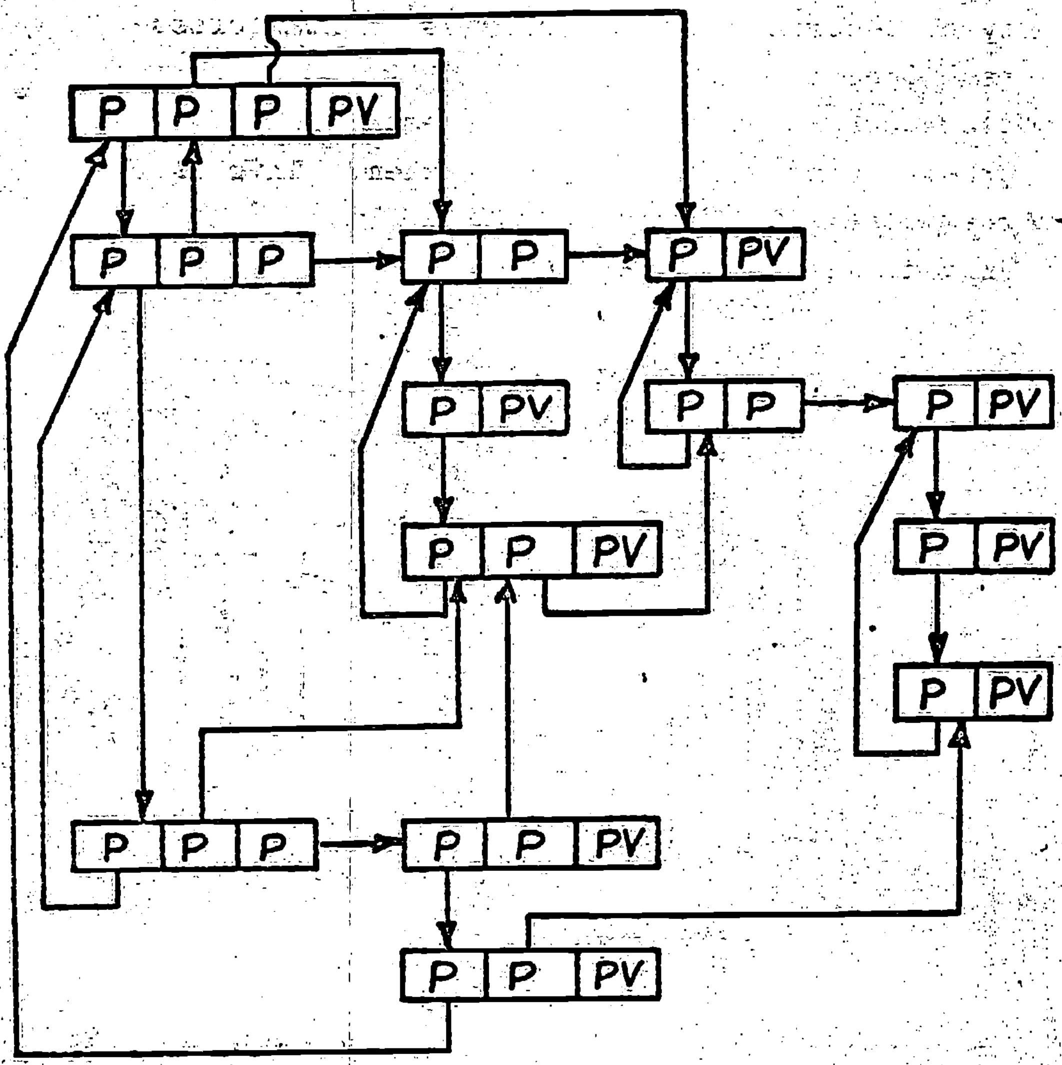


FIGURE 33

PLEX STRUCTURE SCHEMATIC

(Figure reproduced from Kreitner, Ref. 120)

A further refinement is provided by ring structures, Figs. 32, and 33. By connecting head and tail of each branch and by cross-connecting records in different branches any desired path through the network may be described for a search as logical cross-associations may suggest. The purpose is that the programmer may simply declare these logical relations among records instead of having to program procedures to express them. This is the advantage over simple list structures.

For further details, see Refs. 120 and 121.

6.4 References 118 through 121.

118. Roos, D., "ICES, Systems Design—Objectives, Requirements, and Components," Department of Civil Engineering, Massachusetts Institute of Technology, August 1965, reprinted in Ref. 36.

119. Maiorana, C., "COMRADE—Computer-Aided Design Environment," Naval Ship Research and Development Center, paper number AML-61-67, excerpts reprinted in Ref. 36.

120. Kreitner, C. W., "The Universal Data Management System," paper written at American University, excerpts reprinted in Ref. 36.

121. Metaxides, A., "Data Base Management," Bell Telephone Laboratories, Holmdel, New Jersey, reprinted in Ref. 36.

7. Simulation.

7.1 Introduction.

Simulation studies the behavior of a system with the aid of a mathematical model. The real-life system may be too difficult, too expensive, or too time-consuming to test. The system may be any group of objects or activities. It may be a physical, economic, or production system or any other kind of operation or process.

The mathematical analog is a necessary ingredient in any simulation. But to obtain a solution it may be convenient to resort to further analogs governed by the same mathematical model.

For example, an electronic analog may exist (analog computer simulations). Or even a physical analog to a physical problem differing only in scale (model testing). We want to confine the following to cases where the mathematical model is treated by digital methods (digital computer simulation).

In many cases where a mathematical model exists a solution to the problem can also be obtained directly either in a closed form, or at least by approximate methods. Simulation may be superior to this approach under the following circumstances:

*Where the complexity of the system makes direct mathematical treatment difficult.

*Where the system is to undergo many parametric changes such as systematic variation of design variables.

*Where the element of uncertainty is important (stochastic processes). While direct treatment by stochastic operators is not out of the question simulation by Monte Carlo techniques helps to better visualize the problem.

*Where human decisions are to interact with the system (strategic games, decision games, but even design).

*Where a general purpose simulation computer model exists while a special purpose program would have to be developed.

For a thorough discussion of the nature of simulation and a discussion of modeling aspects, see Flanigan's notes in Ref. 124. On queueing, see Ref. 123.

7.2 GPSS.

GPSS (General Purpose Simulation System) is presented as an example for a unique computer language designed for system simulation. The coverage in this course is based on Ref. 122 by Schriber.

7.3 Application to marine systems.

In marine systems many dynamic aspects exist which in principle can be treated by simulation. The dynamics of the ship and its subsystems suggest simulation of physical processes:

*Ship motions

*Maneuvering

*Ship vibrations

*Propulsion system dynamics

However, in most of these cases direct analysis, model studies, and analog computer simulation are the rule, and digital computer simulation the exception.

The emphasis may change gradually as special simulation programs become available, at least where stochastic elements or human decisions are involved.

The true domain of simulation is, however, the area of marine systems operation. Cargohandling, terminal and port operations, navigation, weather routing, etc., are typical examples of complex operations under uncertain conditions.

Fig. 34 shows the block diagram of a complex bulk carrier evaluation. Although for many purposes the analysis of this system may be performed by deterministic methods there are some stochastic elements in the cargo, voyage, and terminal inputs which suggest simulation. Related questions might be:

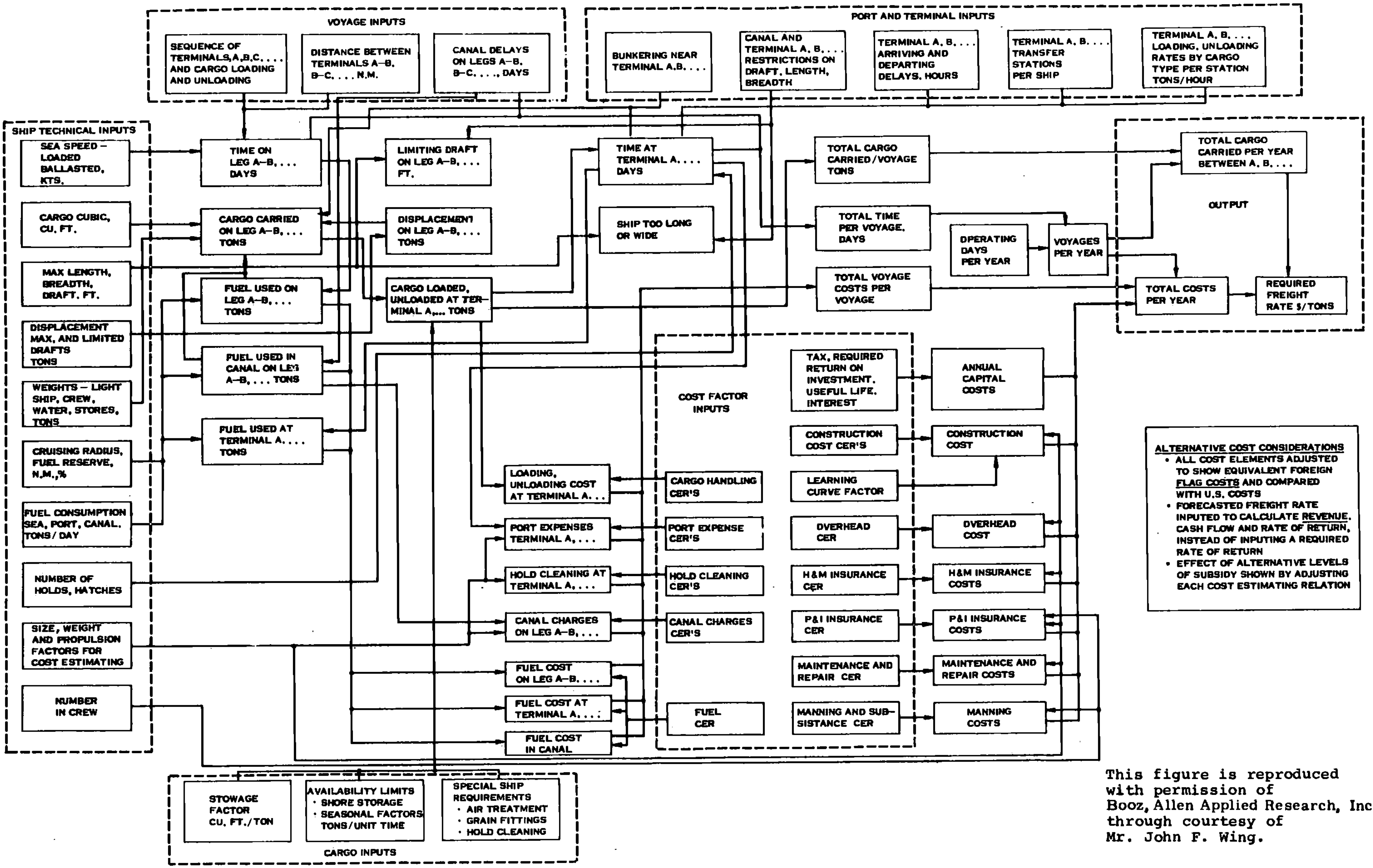
*The effect of voyage delays on the operation

*The effect of fluctuations in cargo availability

*The effect of the terminal service level on system economics

*Most economical level of service relative to ship speed and cargohandling.

It is interesting to note that ship design and operations are tied together inseparably by the economics of the system. This makes the operations simulation a legitimate design tool. In this area, like in many other applications of a computer-aided ship design, we are only at the beginning.


VOYAGE INPUTS
SEQUENCE OF TERMINALS,A,B,C, . . . AND CARGO LOADING AND UNLOADING
DISTANCE BETWEEN TERMINALS A–B, B–C, . . . N.M.
CANAL DELAYS ON LEGS A–B, B–C, . . ., DAYS
PORT AND TERMINAL INPUTS
BUNKERING NEAR TERMINAL A,B, . . .
CANAL AND TERMINAL A, B, . . . RESTRICTIONS ON DRAFT, LENGTH, BREADTH
TERMINAL A, B, . . . ARRIVING AND DEPARTING DELAYS, HOURS
TERMINAL A, B, . . . TRANSFER STATIONS PER SHIP
TERMINAL A, B, . . . LOADING, UNLOADING RATES BY CARGO TYPE PER STATION TONS/HOUR
SHIP TECHNICAL INPUTS
SEA SPEED – LOADED BALLASTED, KTS.
CARGO CUBIC, CU. FT.
MAX LENGTH, BREADTH, DRAFT, FT.
DISPLACEMENT MAX, AND LIMITED DRAFTS TONS
WEIGHTS – LIGHT SHIP, CREW, WATER, STORES, TONS
CRUISING RADIUS, FUEL RESERVE, N.M.,%
FUEL CONSUMPTION SEA, PORT, CANAL, TONS/DAY
NUMBER OF HOLDS, HATCHES
SIZE, WEIGHT AND PROPULSION FACTORS FOR COST ESTIMATING
NUMBER IN CREW
TIME ON LEG A–B, . . . DAYS
CARGO CARRIED ON LEG A–B, . . . TONS
FUEL USED ON LEG A–B, . . . TONS
FUEL USED IN CANAL ON LEG A–B, . . . TONS
FUEL USED AT TERMINAL A, . . . TONS
LIMITING DRAFT ON LEG A–B, . . . FT.
DISPLACEMENT ON LEG A–B, . . . TONS
CARGO LOADED, UNLOADED AT TERMINAL A,... TONS
TIME AT TERMINAL A, . . . DAYS
SHIP TOO LONG OR WIDE
TOTAL CARGO CARRIED/VOYAGE TONS
TOTAL TIME PER VOYAGE, DAYS
TOTAL VOYAGE COSTS PER VOYAGE
OPERATING DAYS PER YEAR
VOYAGES PER YEAR
OUTPUT
TOTAL CARGO CARRIED PER YEAR BETWEEN A, B, . . .
TOTAL COSTS PER YEAR
REQUIRED FREIGHT RATE $/TONS
COST FACTOR INPUTS
TAX, REQUIRED RETURN ON INVESTMENT, USEFUL LIFE, INTEREST
CONSTRUCTION COST CER'S
LEARNING CURVE FACTOR
OVERHEAD CER
H&M INSURANCE CER
P&I INSURANCE CER
MAINTENANCE AND REPAIR CER
MANNING AND SUB-SISTANCE CER
CARGO HANDLING CER'S
PORT EXPENSE CER'S
HOLD CLEANING CER'S
CANAL CHARGES CER'S
FUEL CER
ANNUAL CAPITAL COSTS
CONSTRUCTION COST
OVERHEAD COST
H&M INSURANCE COSTS
P&I INSURANCE COSTS
MAINTENANCE AND REPAIR COSTS
MANNING COSTS
LOADING, UNLOADING COST AT TERMINAL A. . .
PORT EXPENSES TERMINAL A, . . .
HOLD CLEANING AT TERMINAL A, . . .
CANAL CHARGES ON LEG A–B, . . .
FUEL COST ON LEG A–B, . . .
FUEL COST AT TERMINAL A, . . .
FUEL COST IN CANAL
ALTERNATIVE COST CONSIDERATIONS
• ALL COST ELEMENTS ADJUSTED TO SHOW EQUIVALENT FOREIGN FLAG COSTS AND COMPARED WITH U.S. COSTS
• FORECASTED FREIGHT RATE INPUTED TO CALCULATE REVENUE, CASH FLOW AND RATE OF RETURN, INSTEAD OF INPUTING A REQUIRED RATE OF RETURN
• EFFECT OF ALTERNATIVE LEVELS OF SUBSIDY SHOWN BY ADJUSTING EACH COST ESTIMATING RELATION
CARGO INPUTS
STOWAGE FACTOR CU. FT./TON
AVAILABILITY LIMITS
· SHORE STORAGE
· SEASONAL FACTORS
TONS/UNIT TIME
SPECIAL SHIP REQUIREMENTS
· AIR TREATMENT
· GRAIN FITTINGS
· HOLD CLEANING

This figure is reproduced
with permission of
Booz, Allen Applied Research, Inc.
through courtesy of
Mr. John F. Wing.

Fig. 34: Competitive Merchant Ship Project
Bulk Carrier Evaluation Model

7.4 References 122 through 125.

122. Schriber, T. J., "Notes on General Purpose Simulation System/360," The Graduate School of Business Administration, The University of Michigan, 1968, reprinted in Ref. 77.

123. Prabhu, N. U., "Queues and Inventories," John Wiley & Sons, Inc., New York, 1965.

124. Arden, B., et al., "Computer and Program Organization," Lecture Notes, Engineering Summer Conferences, The University of Michigan, 1968.

125. Mettam, J. D., "Forecasting Delays to Ships in Port," The Dock and Harbour Authority, April 1967.

8. Acknowledgments.

It is a pleasure to acknowledge all the help, stimulation, and advice received during the preparation of these notes. The following contributors are mentioned with particular gratitude:

*Professor Johannes Moe, Technical University of Norway, Trondheim, who spent a fruitful year with us in Ann Arbor in 1967/68. The approach he and his staff in Trondheim have been taking to design optimization has been most stimulating and has furnished much background for these notes.

*Professor Brice Carnahan, The University of Michigan, whose excellent teaching and publications on optimization have provided me with many fundamentals of design methodology.

*All other collaborators of our 1968 intensive summer course on Computer-Aided Ship Design, whose notes contain an abundance of useful information: Dr. R. S. Johnson, NAVSEC; Dr. B. Romberg, A. D. Little; Messrs. O. Stephans and C. Maiorana, NSRDC; Mr. S. H. Chasen, Lockheed-Georgia; Professors B. Herzog, T. F. Ogilvie, and F. Westervelt, The University of Michigan.

*The staff of Bethlehem Steel Corp., Sparrows Point, Md., notably Messrs. A. D. Haff, T. F. Robinson, R. V. Danielson, and E. Sweeney who discussed with me the industry's viewpoint of computer-aided design. We also gratefully acknowledge a donation made by Bethlehem Steel Corp. to the Department of Naval Architecture and Marine Engineering. This donation supported a project in computer-aided design paralleling, and thereby benefitting, the preparation of these notes.

*Advice and supporting material was communicated freely from several sources:

Dr. D. Gospodnetić, National Research Council, Ship Division, Ottawa, Canada.

Dr. E. Kantorowitz, Danish Ship Research Institute, Lyngby, Denmark.

Mr. R. C. Kielpinski, Baldwin Kongsberg Company, Cincinnati, Ohio.

Director Sverre Møller, A. S. Bergens Mekaniske Verksteder, Bergen, Norway.

Mr. R. D. Murphy, Department of Transportation, Washington, D.C.
Mr. John F. Wing, Booz, Allen Applied Research, Inc., Bethesda, Maryland.

*Mr. sincere thanks are owed to Miss Kitty Todd for her patient and excellent work in typing the manuscript.

*Finally, I gratefully acknowledge the stimulating interest of my students which made this job worthwhile.